Studienabschlussarbeiten
Fakultät für Mathematik, Informatik
und Statistik

Hubert, Stephanie:

A comparison of multivariate methods for the detection of Differential Item Functioning

Masterarbeit, Wintersemester 2017

Fakultät für Mathematik, Informatik und Statistik

Ludwig-Maximilians-Universität München

https://doi.org/10.5282/ubm/epub.38407

Department of Statistics
Ludwig-Maximilians-University Munich

Master's Thesis

A comparison of multivariate methods for the detection of Differential Item Functioning

Author: Stephanie Hubert
Supervisor: Dr. Gunther Schauberger
Submission date: January 25, 2017

Abstract

This thesis focuses on the comparison of three, recently developed, methods for the detection of differential item functioning (DIF) in the measurement of latent traits, such as abilities or attitudes in psychological or educational research. Identifying group differences is crucial for the correct and unbiased assessment of questionnaires.

Over time, various methods have been proposed in literature to identify test items where DIF is present, which range from test statistics to modeling approaches. Most of these methods have some drawbacks in terms of usability or underlying assumptions, e.g. that they cannot deal with multi-categorical variables or that they focus on the global test level and do not identify DIF on the item level.

The methods presented in this thesis, however, represent an advancement in the sense, that they try to overcome these problems and limitations. A commonality of the methods, that are described in the following, is, that they can cope with both multiple, potentially DIF-inducing, variables and any form of predictor variables, either metric or categorical. The advantage is a flexible and less restricted approach for the detection of DIF.

The first considered method is called *DIFlasso* and is based on an extension of the widely-known Rasch model, that involves additional group-specific parameters to incorporate group differences. DIF-detection is performed using a penalized estimation approach. The second method, *DIFboost*, uses boosting techniques to determine additional group-specific parameters in the extended Rasch model by means of iterative updating of so-called base learners. The third approach called *DIFtree* relies on model based recursive partitioning resulting in a decision tree for every item that carries out DIF.

The aim of this thesis is to compare the three methods regarding their methodological approaches and by means of both an extensive simulation study and an applied example.

Contents

1. Introduction

Latent trait modeling is a common research problem in social or behavioral sciences, when the constructs, variables or attributes of interest are not directly observable but have to be inferred from responses to a set of questions or test items designed for the approximation of the latent trait. Two different measurement frameworks exist: classical test theory (CTT) and the more recent model-based item response theory (IRT). CTT focuses on the global test level and is based on defining the test result of a person as function of the true test score and some error term. IRT models are based on the idea that the probability of a correct response to a test question is determined by the relationship between the individual's ability level and the level of difficulty of the item. Different IRT models exist. Birnbaum (1968) developed the so-called three-parameter logistic (3PL) model, that, in addition to the item difficulty, involves two other item parameters. One allows the items to have different discriminatory power and the other is a pseudo-guessing parameter, that accounts for the fact that in some cases a correct response can be given by simple guessing. The Rasch model, another IRT model developed by Rasch (1960), does not take these two additional parameters into account, relying exclusively on an item difficulty and a person ability parameter. It can be seen as a special case of the 3PL model, even though the two models were developed independently from a different point of origin.

The Rasch model is considered as a basis for all of the methods presented in this thesis and is introduced in the following chapter. The second section of chapter 2 is dedicated to the phenomenon of differential item functioning (DIF). Its detection is the primary goal of the introduced procedures. DIF is present, when the probability of a correct response for two test takers with the same individual ability is not equal, but varies depending on their group membership. Groups can be formed by gender, race, social status, etc.. For more information on DIF in general, see for example Millsap and Everson (1993). Tutz and Schauberger (2015) introduce the DIF model, an extended version of the Rasch model, that can incorporate those group differences and that serves as the underlying model for all the presented methods, introduced in the subsequent chapters, that were developed for an efficient and flexible model-based detection of differential item functioning. Chapter 3 describes the *DIFlasso* procedure (Tutz and Schauberger, 2015), where the DIF model is estimated using lasso penalization. It is followed by a chapter on the *DIFboost* methodology (Schauberger and Tutz, 2016), where parameters for group-specific differences are found via boosting. Chapter 5 first gives an introduction of tree-based modeling in general and for the detection of DIF in particular. Then, two existing concepts, Rasch trees (Strobl et al., 2015) and item focussed trees (*DIFtree*) (Tutz and Berger, 2016; Berger and Tutz, 2016), are explained, whereas the focus is on the *DIFtree* procedure. Here, a tree is grown for every item containing DIF.

A variety of other methods for the detection of DIF has been proposed in literature over time, for a concise overview, see for example Magis et al. (2011). The methods presented here have in common that they were developed to overcome the limitations of existing methods regarding the type and number of predictor variables that can be included. Also, they detect DIF not only on the global test level but on the item level, allowing a conclusion about which items exhibit group differences. Each of the methods captured in this thesis

was compared to the well-established methods in the respective introducing papers and it was shown that they can compete with the established methods regarding their ability to detect DIF. The aim here, however, is to compare the three methods among themselves, both in theory as well as their practical performance, for a broader understanding. In order to assess the practical performance, a simulation study with two different scenarios and different strength of DIF is conducted and covered in chapter 6. Chapter 7 contains an applied example, using a data set that assesses the mathematical abilities of 8th grade students in Austria. This should give further insights regarding the practical performance of the methods and how results differ between the methods in praxis. The thesis concludes with a summarizing comparison of the three methods and an outlook in chapter 8.

2. Basic concepts

This chapter introduces the Rasch model, together with its model requirements and clarifies the meaning of differential item functioning. Different types of DIF are described and established methods that can be used for its detection.

2.1. The Rasch model

The Rasch model (Rasch, 1960) is a popular, widely-used model that provides an estimate for a person's probability of a correct response on a test question.

2.1.1. Model equation

The probability of a person p, $p = 1, ..., P$ correctly solving item i, $i = 1, ..., I$ is specified by:

$$P(Y_{pi} = 1) = \frac{\exp(\theta_p - \beta_i)}{1 + \exp(\theta_p - \beta_i)} \tag{2.1}$$

Y_{pi} denotes a dichotomous variable that indicates whether person p solved item i correctly ($Y_{pi} = 1$) or not ($Y_{pi} = 0$). The probability of a correct response, $P(Y_{pi} = 1)$, depends on two parameters: the person parameter θ_p, that represents the person ability and the item parameter β_i, that denotes the item difficulty. The person parameter varies over persons and the item parameter over items, respectively. The model is not identifiable. Therefore, one parameter (either a person or an item parameter) is commonly set to zero.
The Rasch model can alternatively be expressed in Logit notation:

$$\log\left(\frac{P(Y_{pi} = 1)}{P(Y_{pi} = 0)}\right) = \eta_{pi} = \theta_p - \beta_i \tag{2.2}$$

A commonly used tool to visualize the relation between the latent trait and the probability of correctly solving an item in Rasch models are item characteristic curves (ICCs). The x-axis usually corresponds to the person parameter θ and the y-axis displays the probability of solving an item correctly. Then, the probability of a correct answer according to different person probabilities for different items (with different item difficulties β_i) can be represented graphically. This is exemplarily shown for three fictitious items in figure 2.1. The three curves represent three different items with three different item difficulties. The higher the item difficulty the harder is the item to solve. Here, the item difficulty increases from left to right, meaning that item 1 is the easiest to solve and item 3 the most difficult item. This can be seen from the plot, if one takes a fixed person ability θ and reads off the respective probabilities of solving the item correctly on the y-axis. For example, a person with a personal ability θ of zero would have a 50 percent-chance to get item 2 (with item difficulty $\beta_2 = 0$) right.

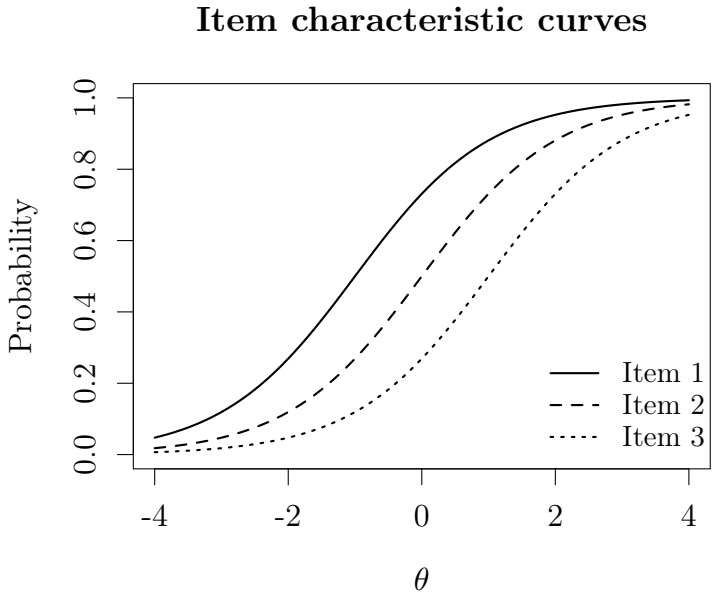

Figure 2.1.: Item characteristic curves for different items with different item difficulties (Item 1: $\beta_1 = -1$, Item 2: $\beta_2 = 0$, Item 3: $\beta_3 = 1$)

2.1.2. Model requirements

Using the Rasch model for the identification of a latent trait, the data should be collected and organized such that five main requirements are fulfilled or at least sufficiently approximated. These are, see Strobl (2012) or Lord (1980):

1. The marginal frequencies of the response matrix of a test are **sufficient statistics** for the person ability and the item difficulty. In general, a statistic is sufficient if it contains all the information about the quantity, that it is supposed to measure. Here, the total score of a person $S_p = \sum_i Y_{pi}$ (the number of correctly solved items of person p) is assumed to be a sufficient statistic for the person ability θ_p. Analogously, $R_i = \sum_p Y_{pi}$ (the number of persons that solve item i) is a sufficient statistic for the item difficulty β_i.

2. **Local stochastic independence**: the probability of solving an item does not depend on the ability to solve another item or on the fact that another person is able to solve the item. Consequently, the overall probability of solving all items is the product of the probabilities of the individual items.

3. **Specific objectivity:** For the comparison of two persons, it does not matter which items (from the pool of possible items) are used. This should hold in the same manner for the comparison of two items, in the sense that it does not depend on which two persons are used for this purpose.

4. **Unidimensionality** presumes that all items of the questionnaire measure one single dimension, one latent trait. Hence, person and item parameters are located on the same latent dimension. An achievement test for example, that requires mathematical

skills and writing skills would not be uni-dimensional in that sense and the use of multidimensional item response theory models would be required.

5. **Monotonicity** holds if with an increase of personal ability the probability for solving an item increases continuously as well. In other words, monotonicity implies a strictly increasing ICC, that is neither decreasing nor constant at any point of the ability interval.

2.2. Differential Item Functioning (DIF)

2.2.1. Definition of DIF

Differential item functioning, also formerly known as item bias, describes the phenomenon of differing probabilities for correctly solving an item among equally able individuals, depending on their group membership. Accordingly, without the presence of DIF, "people of the same ability or skill would have exactly the same chance of getting the item right, regardless of their group membership" as expressed by Lord (1980). Group differences occur for various reasons, e.g. due to cultural or socio-demographic differences between the respondents, such as gender, race, age or religion. Neglecting those differences could lead to misinterpretations of test results and possibly, depending on the purpose of the test, unfair conclusions, see for example Millsap and Everson (1993).

2.2.2. Types of DIF

There are two different types of differential item functioning: uniform and non-uniform DIF. Uniform DIF is present, when the probability of a correct answer in one group is higher than the probability in the other group, independent of the person abilities. Figure 2.2 shows the ICCs for uniform and non-uniform DIF. For uniform DIF (left figure), the ICC of group 1 is constantly above the ICC of group 2. If the ICCs cross each other at some point (right figure), we speak of non-uniform DIF. This means that for some person abilities the probability of solving an item correctly is higher for one group, whereas for other person abilities it is the opposite way. In this thesis the focus is on uniform DIF mainly.

2.2.3. Established methods for the detection of (uniform) DIF

A variety of methods for the detection of DIF has been proposed in literature over time, some of which are IRT-based and some others that are not. All methods have in common, that they were developed in the context of two-group comparisons, where a reference group behaves differently than a focal group. Gender is the classical example of such a binary group comparison. The most prevalent methods are (see Magis et al. (2011) for a good overview):

1. **Mantel-Haenszel procedure:** The Mantel-Haenszel procedure is a non-IRT-based method, that originates from Mantel and Haenszel (1959). It is used to test whether there is a relation between the group membership of a person and their test answers, given the total test score. Let s, $s = 0, ..., I$ denote the number of correctly solved

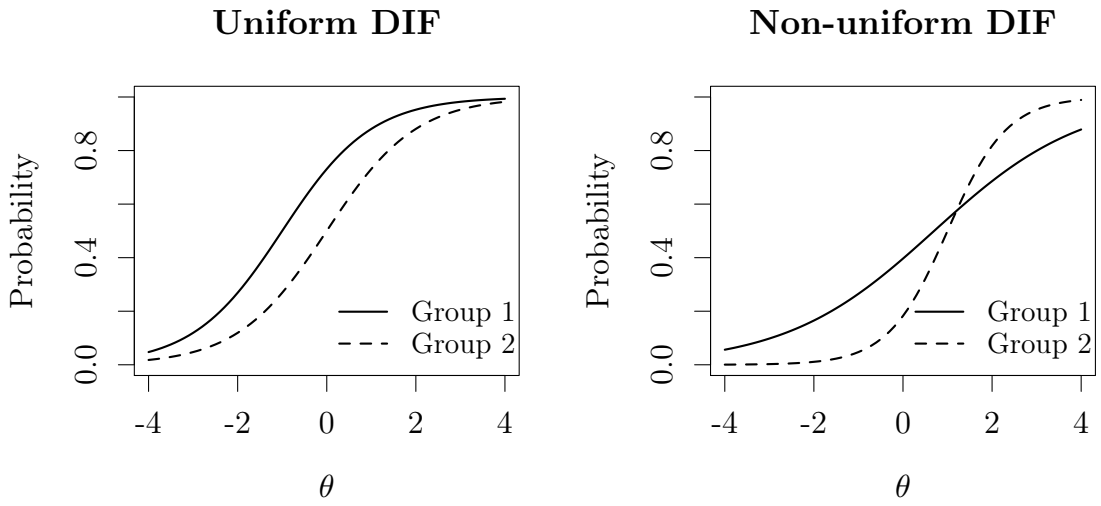

Figure 2.2.: Item characteristic curves for uniform DIF (left) and non-uniform DIF (right)

items for a person and T_s be the total number of persons with respective test score s. Then, for an arbitrary item i, the contingency table for test score s can be written as:

	Item Score s		
	Right answer (R)	Wrong answer(W)	Total
Group 1	A_s	B_s	N_{1s}
Group 2	C_s	D_s	N_{2s}
Groups combined	M_{Rs}	M_{Ws}	T_s

Table 2.1.: Two-by-two contingency table for the Mantel-Haenszel procedure for an arbitrary item i and test score s

The Mantel-Haenszel test statistic for an item i is computed as follows, taking into account all different test scores s:

$$\text{MH} = \frac{\left(|\sum_{s=0}^{I} A_s - \sum_{s=0}^{I} \mathbb{E}(A_s)| - 0.5\right)^2}{\sum_{s=0}^{I} Var(A_s)} \tag{2.3}$$
$$\text{with } \mathbb{E}(A_s) = \frac{N_{1s} M_{Rs}}{T_s} \ , \ Var(A_s) = \frac{N_{1s} N_{2s} M_{Rs} M_{Ws}}{T_s^2 (T_s - 1)}$$

Under the null hypothesis of no difference between group 1 and group 2 for item i, the MH test statistic is asymptotically χ^2-distributed with one degree of freedom. IF MH is larger than a critical calue based on the asymptotic null distribution, DIF is said to occur for item i.

In addition to this, a second test statistic based on the same contingency tables exists: it compares the odds (the ratio of correct and incorrect answers for an item) of two groups, building an odds-ratio across all test scores s:

$$\alpha_{MH} = \frac{\sum_{s=0}^{I} A_s D_s / T_s}{\sum_{s=0}^{I} B_s C_s / T_s} \tag{2.4}$$

Then, $log(\alpha_{MH})$ is approximately normally distributed under the same null hypothesis as above. Values around zero indicate that there is no difference between the odds of the two groups and therefore, that no DIF is present.
The Mantel-Haenszel procedure has been extended to include multiple group comparisons, see Penfield (2001).

2. The **Logistic regression approach** was first introduced by Swaminathan and Rogers (1990) as a cost-effective alternative to IRT-based methods. It can incorporate uniform and non-uniform DIF. The probability of a person p solving an item i is described as a function of the test score $S_p \in 0, ..., I$ (serving as a proxy for the person abilities), the group membership and a possible interaction between the test score and the group membership:

$$log\left(\frac{P(Y_{pi}=1|S_p,g)}{P(Y_{pi}=0|S_p,g)}\right) = \eta_{pi} = \beta_{0i} + S_p\beta_i + \gamma_{ig} + S_p\alpha_{ig} \tag{2.5}$$

g denotes the group, β_{0i} the item-specific intercept, that represents the item difficulties. β_i is the slope of item i. γ_{ig} is a group specific parameter, that comes into play when group specific differences occur. Lastly, $S_p\alpha_{ig}$ denotes the interaction term between the individual test scores and the group membership or, in other words, a group-specific slope. Therefore, γ_{ig} and α_{ig} are the parameters that account for DIF in the logistic regression approach. Uniform DIF is present, if $\gamma_i \neq 0$ and $\alpha_i = 0$, whereas non-uniform DIF is said to occur if $\alpha_i \neq 0$ independent of whether γ_{ig} equals zero or not. Accordingly, if both parameters are zero, no DIF is found. This can be tested by means of a Wald or a likelihood ratio test (Magis et al., 2011).
One version of *DIFtree*, a tree-based method for the detection of DIF, introduced in the following, uses the logistic regression approach as a basis, but incorporates it into a tree-framework. In this context, an extension of the logistic regression model to the multi-group case is considered.

3. **Lord's χ^2-test** (Lord, 1980) can be used to test for group parameters in any IRT model with item discrimination, item difficulty and pseudo-guessing parameters. For a Rasch model, relying on item difficulty parameter β_i exclusively, the test statistic for item i reduces to:

$$\chi^2_{i,Rasch} = \frac{\beta_{1i} - \beta_{2i}}{\hat{\sigma}^2_{1i} + \hat{\sigma}^2_{2i}} \tag{2.6}$$

where β_{1i} and β_{2i} are the item difficulties of group 1 and group 2, and $\hat{\sigma}^2_{1i}, \hat{\sigma}^2_{2i}$ the estimated standard errors of the item difficulties of the two groups. The test statistic is used to test the hypothesis of equal item parameters in both groups. Under the null hypothesis, $\chi^2_{i,Rasch}$ follows an asymptotic χ^2 distribution, where the degrees of freedom correspond to the number of estimated parameters in the model (Magis et al., 2011).
As for the other presented methods, an extended version for the multi-group case exist, see Kim et al. (1995).

The next chapters introduce three recently developed methods, that provide an alternative and more flexible model-based detection of differential item functioning, starting with the *DIFlasso* procedure.

3. DIFlasso

The *DIFlasso* procedure is based on an extension of the Rasch model. For simplicity, this extension will be called DIF model in the following. DIF is represented by additional group parameters $\boldsymbol{\gamma}$ for every item. Via penalization of the group parameters during the joint likelihood estimation, those group parameters are set to zero when no DIF is present, which allows a conclusion about DIF-free and DIF-containing items.

3.1. Model equation

To incorporate group-specific differences, the simple Rasch model (cf. eq. (2.2)) is extended to include the term $\mathbf{x}_p^T \boldsymbol{\gamma}_i$ (Tutz and Schauberger, 2015):

$$\log\left(\frac{P(Y_{pi}=1)}{P(Y_{pi}=0)}\right) = \theta_p - (\beta_i + \mathbf{x}_p^T \boldsymbol{\gamma}_i) \tag{3.1}$$

with $\mathbf{x}_p^T$ being the person-specific covariate vector for person p and $\boldsymbol{\gamma}_i$ being the item-specific vector of group parameters for item i. If $\boldsymbol{\gamma}_i = 0$ for an item i, the item is considered to be DIF-free and model (3.1) reduces to the simple Rasch model. If $\boldsymbol{\gamma}_i \neq 0$ for an item i, the item is regarded as a DIF-item and thus, the overall item difficulty of item i, β_i, is complemented by the term $\boldsymbol{\gamma}_i$, depending on the group membership of person p. Together, these two terms form the individual, person-specific item difficulty for person p and item i.

In the simplest case of one binary covariate, say gender, with male being the reference category, the person-specific item difficulties for item i are β_i for males and $\beta_i + \gamma_i$ for females. Analogously one could also apply effect coding instead of reference coding. Then, β_i remains the overall item difficulty as known from the simple Rasch model and $\beta_i - \gamma_i$ for males and $\beta_i + \gamma_i$ for females would be the person-specific item difficulties, which are located symmetrically around the overall item difficulty β_i.

This concept can easily be generalized. For a categorical variable with k categories $\boldsymbol{\gamma}_i$ is of length k-1. Again, β_i is the person-specific item difficulty for reference category k and $\beta_i + \gamma_{ik}$ define the person-specific item difficulties for every category $1, ..., k-1$. For a metric covariate, say age, one γ_i is estimated and the person-specific item difficulty is given by $\beta_i + \text{age}_p\ \gamma_i$, depending on the value of the covariate for person p (here the age of person p).

3.2. Penalization for the detection of DIF

The motivation behind a penalization of parameters varies from setting to setting. In high dimensional settings, the number of parameters gets very large which can lead to unstable parameter estimates or sometimes, parameters cannot be estimated at all. This makes regular maximum likelihood estimation problematic. Here, the motivation behind penalization is a different. Parameters are penalized in order to detect DIF items and variables. Consequently, penalization is only applied to the group parameters $\boldsymbol{\gamma}$ and the person and item parameters are estimated regularly. Then, DIF is assumed to occur when not all the entries of the group parameter vector γ for an item are shrunk to zero.

Instead of maximizing the log-likelihood $l(\boldsymbol{\alpha})$ with $\boldsymbol{\alpha}^T = (\boldsymbol{\theta}^T, \boldsymbol{\beta}^T, \boldsymbol{\gamma}_1^T, ..., \boldsymbol{\gamma}_I^T)$ during the estimation process, a penalized version of the log-likelihood is maximized:

$$l_p(\boldsymbol{\alpha}) = l(\boldsymbol{\alpha}) - \lambda J(\boldsymbol{\alpha}) \tag{3.2}$$

with $J(\boldsymbol{\alpha})$ being a penalty term that penalizes certain structures in the parameter vector and λ regulating the strength of the penalty term. The smaller the value of λ the smaller is the penalization. In the extreme case of $\lambda = 0$, the penalty term drops out and the regular likelihood is maximized. For $\lambda \to \infty$, respectively large enough, all the parameters associated with the penalty term are shrunk to zero. λ is a tuning parameter. The choice of an optimal λ is covered in the next paragraph.
A common penalty term is $J(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T\boldsymbol{\alpha}$, the squared length of the parameter vector, which is known as Ridge penalty. The Ridge penalty reduces the size of the parameters. This increases the stability of the parameters, but they cannot be reduced to zero completely. Therefore, no parameter selection is possible. In the context of DIF detection, the lasso (least absolute shrinkage operator) penalty seems conceptually more appropriate. For the *DIFlasso* procedure, it is defined as:

$$J(\boldsymbol{\alpha}) = \sum_{i=1}^{I} |\boldsymbol{\gamma}_i| \text{ with } |\boldsymbol{\gamma}_i| = (|\gamma_{i1}| + ... + |\gamma_{im}|) \tag{3.3}$$

Here, the absolute length of the group parameter vector is penalized. Each parameter of a $\boldsymbol{\gamma}_i$ is treated independently. If one γ_{ij} is unequal to zero, DIF is said to occur for the respective item i and according to variable j. The penalty term includes the group-specific parameters only. Person and item parameters are left out of the penalization and are fully included into the model.
Yuan and Lin (2006) introduced a general group lasso penalty for situations where penalizing groups of parameters seems more appropriate than individual penalization. Tutz and Schauberger (2015) present a version of the group lasso penalty for *DIFlasso*, where grouping is based according to the items:

$$J(\boldsymbol{\alpha}) = \sum_{i=1}^{I} ||\boldsymbol{\gamma}_i|| \tag{3.4}$$

where $||\boldsymbol{\gamma}_i|| = (\gamma_{i1}^2 + ... + \gamma_{im}^2)^{1/2}$. In this case, all γ-parameters for an item are treated simultaneously and either all γ-parameters for an item are set to zero or none. If $\boldsymbol{\gamma}_i = 0$, the item is classified as free of DIF. If $\boldsymbol{\gamma}_i \neq 0$, DIF is said to occur for the respective item i and in model (3.1), the overall item difficulty β_i is modified by the term $\mathbf{x}_p^T\boldsymbol{\gamma}_i$, resulting in person-specific item difficulties for item i, depending on the group membership of the test takers.

Choice of lambda

λ is a tuning parameter that needs to be chosen carefully, since it determines the choice of the final model. In practice, the optimal λ is derived as a compromise between the sparseness of the model and the model fit. Therefore, a range of λ-values is identified, corresponding to different strengths of the penalization. The DIF model is estimated with

each λ, yielding as many models as there are λ's. Then, in a next step, the "optimal" λ is determined by means of some criterion. Several criteria exist, like the Akaike information criterion (AIC) or the Bayesian information criterion (BIC), which is used here:

$$BIC(\lambda) = -2 \cdot l(\boldsymbol{\alpha}) + \mathrm{df}(\lambda) \cdot \log(P \cdot I) \tag{3.5}$$

with degrees of freedom $df(\lambda) = I + P + \tilde{df}_{\boldsymbol{\gamma}}(\lambda) - 1$ and

$$\tilde{df}_{\boldsymbol{\gamma}}(\lambda) = \sum_{i=1}^{I} I(||\boldsymbol{\gamma}_i(\lambda)|| > 0) + \sum_{i=1}^{I} \frac{||\boldsymbol{\gamma}_i(\lambda)||}{||\boldsymbol{\gamma}_i^{ML}(\lambda)||}(m-1) \tag{3.6}$$

as proposed by Yuan and Lin (2006). The term $\tilde{df}_{\boldsymbol{\gamma}}(\lambda)$ consists of one degree of freedom for every DIF item and a fraction of the number of covariates m minus one, depending on the size of the L2 norm of the γ-parameters with penalization in relation to the L2 norm without penalization. This fraction on itself is also implemented as another version of calculating $\tilde{df}_{\boldsymbol{\gamma}}(\lambda)$, referred to as the L2 norm type of degrees of freedom in the following. The BIC, in general, leads to more parsimonious models than the AIC. The smaller the BIC the better. In the end, the model is chosen as the final model (and corresponding λ) that has the smallest BIC-value of all considered models.
Figure 3.1 shows the course of $||\boldsymbol{\gamma}_i||$ over the different values of λ for two settings (weak and strong DIF) from the following simulations (one randomly chosen iteration of scenario 1). λ ranges from 0 to the values where all $||\boldsymbol{\gamma}_i||$ are shrunk to zero. The larger the value of λ the larger is the penalization. The four DIF items are indicated by the dotted lines. The vertical dashed line marks the BIC-optimal model. For the strong DIF setting, all γ-parameters except for the DIF items are shrunk to zero at that point yielding an optimal detection rate. In the weak DIF setting, where group differences are small, only one of the four DIF items is correctly diagnosed as such at BIC-optimal λ. Again, none of the items is falsely identified as DIF item.

3.3. Identifiability issues

In the simple Rasch model, one item or person parameter has to be set to zero (cf. section 2.1.1) to ensure identifiability of the parameters. The DIF model (3.1) is overparameterized as well, and therefore, parameters are again not identifiable. Reparametrization with a constant vector $\mathbf{c}$ leads to the same model:

$$\begin{aligned} \eta_{pi} = \theta_p - \beta_i - \mathbf{x}_p^T\boldsymbol{\gamma}_i &= \theta_p - \beta_i - \mathbf{x}_p^T(\boldsymbol{\gamma}_i - \mathbf{c}) - \mathbf{x}_p^T\mathbf{c} \\ &= \tilde{\theta}_p - \beta_i - \mathbf{x}_p^T\tilde{\boldsymbol{\gamma}}_i \end{aligned} \tag{3.7}$$

with $\tilde{\theta}_p = \theta_p - \mathbf{x}_p^T\mathbf{c}$ and $\tilde{\boldsymbol{\gamma}}_i = \boldsymbol{\gamma}_i - \mathbf{c}$. Therefore, in addition to the restriction from the Rasch model, one γ-vector has to be set to zero.
During the *DIFlasso* procedure this is realized as follows: After the penalized estimation of the DIF model, item i is identified, whose γ-parameters were shrunk to zero first as the size of the penalty term λ increases. Subsequently, this item is defined as the reference item. The corresponding γ-vector equals zero already. Its item difficulty β_i is afterwards set to zero, which makes the model identifiable and interpretable in relation to the reference item i.

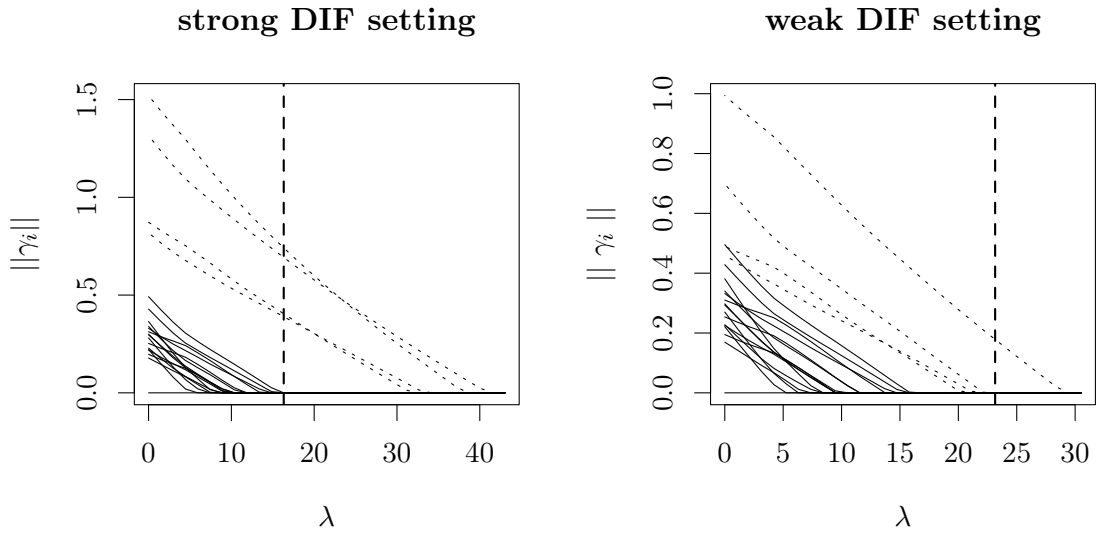

Figure 3.1.: Exemplary visualization of the L2 norm of item-specific parameter estimates over lambda in one iteration of simulation scenario 1

3.4. Model estimation

Different types of maximum likelihood (ML) estimations exist, such as the marginal, conditional or joint ML estimation. Usually, all three of them give comparable results. The *DIFlasso* procedure uses joint ML estimation. All parameters of interest are computed simultaneously. This means that in every step of the iterative procedure, new values are generated for all parameters.

3.5. GLM representation of the DIF model

The DIF model can be embedded into the framework of generalized linear models (GLMs). In software, model estimation via GLMs is a common and well implemented way of model fitting. This is also used for the *DIFlasso* procedure.
The general representation of a GLM is:

$$g(\pi_{pi}) = z_{pi}^T \, \alpha \tag{3.8}$$

where g is a link function that links the outcome to the linear predictor $z_{pi}^T \, \alpha$, containing design matrix z_{pi}^T and vector of parameters α.
The DIF model can be expressed as a GLM with binary response and logit link. Equation (3.1) can be written as, see Tutz and Schauberger (2015):

$$\begin{aligned} g(\pi_{pi}) = \log\left(\frac{P(Y_{pi}=1)}{P(Y_{pi}=0)}\right) &= \theta_p - (\beta_i + \mathbf{x}_p^T \boldsymbol{\gamma}_i) \\ &= \mathbf{1}_{P_{(p)}}^T \boldsymbol{\theta} - \mathbf{1}_{I_{(i)}}^T \boldsymbol{\beta} - \mathbf{x}_p^T \boldsymbol{\gamma}_i \\ &= \mathbf{z}_{pi}^T \boldsymbol{\alpha} \end{aligned} \tag{3.9}$$

Using this notation, $\mathbf{1}^T_{P_{(p)}} = (0, ..., 0, 1, 0, ..., 0)$ and $\mathbf{1}^T_{I_{(i)}} = (0, ..., 0, 1, 0, ..., 0)$ specify vectors of length P resp. (I-1) and equal one at position p resp. i.
$\mathbf{z}^T_{pi} = (\mathbf{1}^T_{P_{(p)}}, -\mathbf{1}^T_{I_{(i)}}, 0, ..., 0, -\mathbf{x}^T_p, 0, ..., 0)$ and $\boldsymbol{\alpha} = (\boldsymbol{\theta}^T, \boldsymbol{\beta}^T, \boldsymbol{\gamma}^T_1, ..., \boldsymbol{\gamma}^T_I)$ describes the parameter vector, as defined above.

For the simplified case of data set with two observations and two items, the response vector $\boldsymbol{y}$, design matrix $\mathbf{Z}$ containing all vectors $\mathbf{z}^T_{pi}$ and parameter vector $\boldsymbol{\alpha}$ would be:

$$y = \begin{pmatrix} y_{11} \\ y_{12} \\ y_{21} \\ y_{22} \end{pmatrix}, \; Z = \begin{pmatrix} 1 & 0 & -1 & -x_1^T \\ 1 & 0 & 0 & 0 \\ 0 & 1 & -1 & -x_2^T \\ 0 & 1 & 0 & 0 \end{pmatrix}, \; \alpha = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \beta_1 \\ \gamma_1 \end{pmatrix}$$

In this notation, $\beta_1 = 0$ and $\gamma_1 = 0$ to ensure the identifiability of the parameters. Therefore, item 2 would be the reference item.

4. DIFboost

Same as *DIFlasso*, the *DIFboost* strategy aims at representing the data as a DIF model. In contrast to *DIFlasso*, DIF parameters are not found via penalized model estimation but via boosting. The idea behind boosting is to combine many "weak" learners to "form a powerful committee" (Friedman et al., 2000). In an iterative procedure, a single weak learner is fitted to the data in every step. The model output of the current iteration does only depend on the output from the previous step and the weak base learner of the current step. In the context of DIF detection, every parameter of the DIF model serves as possible base learner. DIF items are determined by those group-specific parameters that are selected during the boosting algorithm. The final model is then, in a second step, estimated using regular maximum likelihood estimation.

4.1. The concept of boosting

Boosting is a general concept that can be used in many different, especially high-dimensional, settings, where dimension or parameter reduction is of interest. Before linking the concept of boosting to parameter selection in DIF research, this section should give a general introduction. It follows the explanations of Friedman et al. (2000) and Friedman (2001) and starts with an additive model for some quantitative response y, before proceeding to the case where the response is restricted to be in $[0, 1]$, as it is the case in item response modeling. The predictor variables x_j, $j = 1, ..., p$, and response y are supposed to have some joint distribution. Let us consider an additive model, where we are interested in modeling the mean $\mathbb{E}(y|x) = F(x)$. It has the form:

$$F(x) = \sum_{j=1}^{p} f_j(x_j) \tag{4.1}$$

Each of the m predictor variables x_j is included in the model via a function $f_j(x_j)$.
For more general additive models, those functions can be functions of potentially all of the predictor variables x. Then,

$$f_m(x) = \beta_m \ \mathrm{h}(x, \alpha_m) \tag{4.2}$$

where $f_m(x)$, $m = 1, ..., M$, is taken to be a simple function $h(x, \alpha_m)$ characterized by the predictor variables and a set of parameters α_m and a multiplier β_m. The additive model then becomes

$$F_M(x) = \sum_{m=1}^{M} \beta_m \ \mathrm{h}(x, \alpha_m) \tag{4.3}$$

Parameters are usually estimated by minimizing some loss function L(y, F(x)). Boosting can be seen a stagewise algorithm for fitting additive models. Then, for every parameter iteration m=1,...,M

$$(\beta_m, \alpha_m) = \arg\min_{\beta,\alpha} \sum_{i=1}^{N} L\big(y_i, F_m(x_i)\big) \tag{4.4}$$

The current model $F_m(x)$ of boosting step m is composed of $F_{m-1}(x)$, the model of the previous step (m-1) (also referred to as the offset), and the current update of step m:

$$F_m(x) = F_{m-1}(x) + \beta_m \mathrm{h}(x, \alpha_m) \tag{4.5}$$

In this stepwise approach, one parameter is updated in every boosting iteration. The parameter is taken, that yields the greatest reduction in terms of the loss function. The previous terms are not readjusted when a new term enters. For each iteration either a parameter that is already included in the model is updated or a new one is added. The choice of L(y,F(x)) depends on the application and leads to different boosting algorithms. In the boosting terminology $h(x, \alpha_m)$ are called base learners.
Fitting the data too closely can be counterproductive. A way to prevent overfitting is to constrain the number of boosting iterations M. Different concepts for the determination of the best value for M exist, details are given in section 4.3. In addition to that, it is commonly believed that a better fit is achieved when the parameter updates are small. This is realized with an additional shrinkage parameter ν:

$$F_m(x) = F_{m-1}(x) + \nu \cdot \beta_m h(x, \alpha_m) \qquad 0 < \nu \leq 1 \tag{4.6}$$

Each update is scaled by the value of the shrinkage parameter. Its value could be tuned, but is commonly set to $\nu = 0.1$ which is proven to yield sufficiently good results. Both, M and ν control the degree of fit and thus affect each other. For example, increasing the strength of shrinkage increases also the best value for M.

In item response modeling one is interested in the probability of a correct response of a person to a test question. In cases when the response estimates are restricted to be in $[0, 1]$, logistic regression is a popular approach. An additive logistic model has the form:

$$\eta_{pi} = \log\left(\frac{P(Y_{pi} = 1)}{P(Y_{pi} = 0)}\right) = \sum_{m=1}^{M} f_m(x) \tag{4.7}$$

Solving (4.7) for $P(Y_{pi} = 1)$, yields:

$$\pi_{pi} = P(Y_{pi} = 1) = \frac{e^{F(x)}}{1 + e^{F(x)}} \tag{4.8}$$

The procedure of finding parameter updates in every iteration by minimizing a loss function works in the same way as described above. The Rasch model and also the DIF model (3.1) are special cases of this general additive logistic model with components $\theta, -\beta$ (and $-x^T\gamma$ for the DIF model). The appropriate loss function in this case is the negative likelihood of a binomial logit model. In the next section, the estimation of the DIF model using boosting is described in more detail.

4.2. Boosting for the detection of DIF

The *DIFboost* algorithm, proposed by Schauberger and Tutz (2016), proceeds as follows: First, a simple Rasch model is fitted, yielding parameter estimates for the person and item parameters. This ensures that parameter selection refers to DIF effects only and that person and item parameters are always included in the model, as desired.
From the person and item parameter estimates, the linear predictor $\hat{\eta}_{pi} = \hat{\theta}_p - \hat{\beta}_i$ is calculated for each combination of observation and item. These linear predictors from the Rasch model are collected in $\hat{\boldsymbol{\eta}}_{RM} = (\hat{\eta}_{11}, \hat{\eta}_{12}, ..., \hat{\eta}_{IP})$ and are used to initialize the

boosting algorithm. $\hat{\boldsymbol{\eta}}_{RM}$ is called the offset or base model for the boosting procedure and is denoted as $\eta^{(0)}$.
For the boosting steps, the Rasch model is extended to the DIF model that also includes group-specific coefficients. Each of the model components serves as a possible base learner:

$$\tilde{\eta}(\mathbf{x}_p, p, i) = \begin{cases} \tilde{\theta}_p & p = 1, ..., P-1 \\ \tilde{\beta}_i & i = 1, ..., I \\ \mathbf{x}_p^T \tilde{\boldsymbol{\gamma}}_i & i = 1, ..., I \end{cases} \tag{4.9}$$

Initially, before the first boosting iteration, all possible base learners $\tilde{\theta_p}, \tilde{\beta}_i$ and $\tilde{\boldsymbol{\gamma_i}}$ are zero. In every step of the procedure, one single parameter is updated. To determine which base learner that is, one minimizes an adequate loss function $L(Y_{pi}, \tilde{\pi}_{pi})$ for every possible base learner. $\tilde{\pi}_{pi}$ denotes the fitted probability of a person p to solve item i. Here, for a logit model with binary response, the loss function is the negative likelihood of a binomial logit model:

$$L(Y_{pi}, \tilde{\pi}_{pi}) = -(Y_{pi} \log(\tilde{\pi}_{pi}) + (1 - Y_{pi}) \log(1 - \tilde{\pi}_{pi})) \tag{4.10}$$

In every boosting step $m, m = 1, ..., M_{stop}$, the base learner is chosen that yields the greatest reduction of the loss function:

$$\tilde{\eta}^*(\mathbf{x}_p, p, i) = \arg\min_{\tilde{\theta_p}, \tilde{\beta}_i, \mathbf{x}_p^T \tilde{\boldsymbol{\gamma_i}}} \sum_{p,i} L(Y_{pi}, \tilde{\pi}_{pi}) \tag{4.11}$$

Then, the model predictor of the current boosting step m is

$$\tilde{\eta}^{(m)} = \tilde{\eta}^{(m-1)} + \nu \; \tilde{\eta}^*(\mathbf{x}_p, p, i) \tag{4.12}$$

It consists of the predictor of the previous step $(m-1)$ and the currently considered base learner. Parameter ν, $0 < \nu < 1$, regulates the extent to which each predictor $\tilde{\eta}^*$ updates the model. It is used to avoid quick overfitting. To guarantee small step sizes, ν is taken to be sufficiently small (typically $\nu = 0.1$).
Lastly, the predictor $\tilde{\eta}^{(m)}$ is used to calculate the probability $\tilde{\pi}_{pi}$ of the current boosting step:

$$\tilde{\pi}_{pi} = \frac{exp(\tilde{\eta}^{(l)})}{exp(1 + exp(\tilde{\eta}^{(l)}))} \tag{4.13}$$

This procedure of choosing base learners and model updating is repeated for a predefined number of steps M_{stop}.
Boosting in this context is used for the selection of relevant model parameters and can be seen as a method for the detection of DIF regarding the selection of group-specific parameters. In order to determine which parameters are to be included in the final model, one relies on the concept of stability selection, that is described in the following section. Stability selection finds a set of stable parameters by repeating the boosting procedure on subsamples of the original data. The final model from the *DIFboost* procedure includes only these stable parameters. It is estimated in a second step using regular maximum likelihood estimation.

Remarks Even though the person and item parameters of the Rasch model are estimated before the boosting procedure and serve as the base model, they are also included as possible base learners in the boosting algorithm. Since additional group-specific parameters may enter the boosting model, it might become necessary to also update person and item parameters in a later iteration of the boosting algorithm to improve the model fit.
All base learners are linear. But since the different model components contain a different number of model parameters, their chances of being chosen in the boosting iteration are not the same. For example, it would be more likely to choose a person parameter than an item parameter, because usually there are many more observations in the data set than items. To avoid this problem, the degrees of freedom of the base learners can be regulated using internal penalty terms. Here, a Ridge penalty is applied to every base learner to restrict its degree of freedom to one. This keeps the degrees of freedom consistent over the different parameters.

4.3. Stability selection

To prevent the boosting model from overfitting, a stopping criterion should be used, that determines the number of boosting iterations and stops the procedure before the model is possibly overfit after the prespecified M_{stop} boosting iterations. Also, in the context of DIF detection, a main goal is parameter selection in order to determine which items are DIF items. The estimation of a full model without parameter selection is not of interest.
A common way to achieve parameter selection is to stop the boosting procedure after an appropriate number of iterations. This is often called "early stopping". The number of iterations can be determined by crossvalidation techniques or an information criterion.
An alternative to early stopping, proposed by Meinshausen and Bühlmann (2010), is stability selection. The *DIFboost* procedure uses stability selection for the selection of the model parameters. Stability selection is based on subsampling.
First, a random subset of half the size of the data set is drawn. Then, the boosting model is fit on the subset. One does not proceed to the prespecified number of boosting iterations M_{stop}, but stops as soon as q distinct base learners are selected for the subsample. q has to be prespecified as well and is usually taken to be $0.6 \cdot I$, presuming that no more than 60% of the test items contain DIF. If the boosting algorithm proceeds to the maximal number of boosting iterations without finding q base learners, a warning will be displayed and M_{stop} should be increased.
The subsampling and model fitting by boosting is repeated a fixed number of times B. Let $\hat{S}_b$ denote the set of selected base learners in replication b. Then, one computes for every boosting step m the relative frequencies

$$\hat{\Pi}_i^m = \frac{1}{B} \sum_{b=1}^{B} I_{i \in \hat{S}_b, b} \tag{4.14}$$

that, for every base learner i, indicate in how many of the replications the base learner was chosen by the boosting algorithm. A cutoff point π_0 is defined and with the aid of this cutoff point, a set of stable base learners is given by

$$\hat{S}^{stable} = \{i : max_{m=1,\ldots,M_{stop}}(\hat{\Pi}_i^m) \geq \pi_0\} \tag{4.15}$$

This means that a base learner is included in the final model if its relative frequency over all replications is larger than the cutoff point for at least one of the boosting iterations. Meinshausen and Bühlmann (2010) propose to chose $\pi_0 \in (0.6, 0.9)$. Originally, π_0 is a tuning parameter, but values in the given range tend to give very similar results, which is why π_0 is usually fixed.

4.4. Identifiability issues

In section 3.3 it was shown that the DIF model is not identifiable. Additional parameters have to be fixed. But the models are identifiable as long as at least the DIF parameters for one item are not chosen during the boosting procedure, thus one item is DIF-free. In practice, one defines one of the items as reference item, where no DIF parameters were selected during the boosting procedure. Then, $\boldsymbol{\gamma}_R = 0$ for the reference item, by nature. For reasons of simplicity, the additional restriction is $\beta_R = 0$ instead of $\theta_P = 0$.

4.5. The DIFboost algorithm

The *DIFboost* algorithm proceeds as follows:

DIFboost

Step 1 (Initialization)

- Fit the Rasch model for given scores Y_{pi} and initialize the offset $\tilde{\eta}^{(0)} = \hat{\eta}_{RM}$
- Initialize $\tilde{\theta}_p = 0,\ p = 1, ..., P-1,\ \tilde{\beta}_i = 0$ and $\boldsymbol{\gamma}_i = \mathbf{0},\ i = 1, ..., I$
- Set $m = 0$

Step 2 (Iteration)

- $m \to m + 1$
- Fit a logit model for every possible base learner where $\tilde{\eta}^{m-1}$ is used as offset
- Select the best base learner $\eta^*(\mathbf{x}_p, p, i)$
- Update the linear predictor by

$$\tilde{\eta}^{(m)} = \tilde{\eta}^{(m-1)} + \nu\tilde{\eta}^*(\mathbf{x}_p, p, i)$$

Step 3 (Stop)

- Iterate *Step 2* until $m = M_{stop}$ is reached

As described above, the procedure is replicated a fixed number of times relying on the concept of stability selection, resulting in a set of stable base learners. The final model is then estimated via regular maximum likelihood estimation using the set of previously determined base learners.

5. Tree-based DIF modeling

Tree-based modeling can be used for the detection of differential item functioning. The general advantage over other existing methods is that groups do not have to be prespecified. Moreover, in comparison to *DIFlasso* and *DIFboost*, non-linear DIF effects can be captured as well and the tree structure simplifies the detection of interactions between the predictor variables. Different concepts exist: Rasch trees (Strobl et al., 2015) build a single tree for the complete test data. Hence, they indicate whether DIF is present in the test or not and which variables are affected but they do not show the responsible items in a very intuitive way. The *DIFtree* procedure (Tutz and Berger, 2016; Berger and Tutz, 2016) tries to overcome that limitation by building trees on the item level. If no DIF is present, no tree is built for the respective item. In the following, the concept of tree-based modeling in general is briefly introduced, followed by a subsection about Rasch trees. The main part of the section is dedicated to item-focussed trees for the detection of differential item functioning. The terms item focused trees and *DIFtree* thereby specify the same methodological approach and are used interchangeably.

5.1. The concept of tree-based modeling

Tree-based modeling, in general, works as follows: a tree, either a regression or a classification tree, is built by recursively partitioning the feature space (the space spanned by all the predictor variables) into a set of rectangular areas. Each partition is described by a node in the resulting tree. The tree extends from the root node to the terminal nodes. Each of the terminal nodes represents a unique partition in the feature space and in each of the partitions, a simple model, in most cases a constant, is fitted. The two most popular algorithms for tree-based modeling are the CART algorithm (Breiman et al., 1984) and the C4.5 algorithm (Quinlan, 2014).
The partitioning of the feature space is achieved differently, depending on the type of covariate. For metric and ordinal variables, a cutoff point c based on one variable x is chosen. The objects of node A are grouped into the two subcategories according to their value of x in relation to the cutoff point c

$$A \cap \{x \leq c\},\ A \cap \{x > c\}.$$

Node A can represent the total feature space if the first split is considered or any partition of the feature space in further steps of the splitting procedure.
For two-categorical variables, the objects are grouped according to the two categories. For (unordered) categorical variables with k (k≥2) categories, two options exist: the C4.5 algorithm divides the objects into as many groups as there are categories, resulting in k daughter nodes, while the CART algorithm produces binary splits, grouping categories together, if necessary. In the following, the focus is on binary splitting exclusively. Here, the partition of node A has the form

$$A \cap S,\ A \cap \bar{S},$$

where S is a non-empty subset $S \subset \{1, ..., k\}$ and $\bar{S} = \{1, ..., k\} \setminus S$ is the complement.
In each step, the splitting is conducted according to the combination of variable and

split point that performs best in terms of a certain split selection criterion. One idea is to choose the combination of variable and split point yielding the greatest impurity reduction. Several measurements for the quantification of impurity exist, e.g. the Gini index or the Shannon entropy. Other methods for the detection of the optimal split point are test-based, as the concept of maximally selected statistics that is used in the following. It is never advisable to grow trees to their maximal size, as they would most likely overfit and their generalizability becomes questionable. Different concepts exist to determine the size of a tree. Either all splits are performed, the tree is grown to its maximal size and pruning is applied afterwards. Hereby, the tree is cut back to avoid that it is fit too closely to the data. Another possibility is to test the significance of each split during the splitting procedure and stop when no more splits are found to be significant.

Recursive partitioning for the detection of differential item functioning

Recursive partitioning of the feature space can be accomplished based on the values of the response variable. Splitting is done when the outcome of the response variable varies between groups formed by the predictor variables. A more flexible approach is model based recursive partitioning, where splitting is performed when the parameters of a parametric model vary between groups formed by predictor variables. In the context of DIF detection, model-based recursive partitioning is used. DIF is detected, respectively a split is carried out, when the item difficulty parameters of the model vary between subgroups.

Figure 5.1 shows four exemplary trees built by the *DIFtree* procedure in scenario 1 of the simulations, each corresponding to one test item. The trees expand from the root node on top to the terminal nodes. The terminal nodes contain a value for the item difficulty in the corresponding subsample of the data. On each level, the splitting rule (splitting variable and split point) is displayed. Here, the number of DIF variables varies between two and three. Multiple splits can be conducted corresponding to the same variable (see item 2). The number of splits also varies across items. The resulting item difficulties can only be interpreted relatively to each other. A lower value is interpreted as a lower item difficulty and therefore, a higher probability of solving the item correctly.

5.2. Rasch trees

The main motivation behind the concept of Rasch trees, as explained by the authors Strobl et al. (2015), is to provide an easily interpretable representation of DIF, where groups are not pre-specified and thus, to gain an understanding of the psychological sources of differential item functioning. The main difference between Rasch trees and item-focussed trees is the number of trees built. Whereas for Rasch trees a single tree for the whole test is built, in the context of item-focussed trees, as many trees are built as there are DIF items. Therefore, it should be kept in mind that the two concepts serve different purposes: Rasch trees focus on the identification of variables that are responsible for DIF, while item-focussed trees additionally help to also identify affected items.
First, the item parameters of the Rasch model are estimated jointly for the full sample.

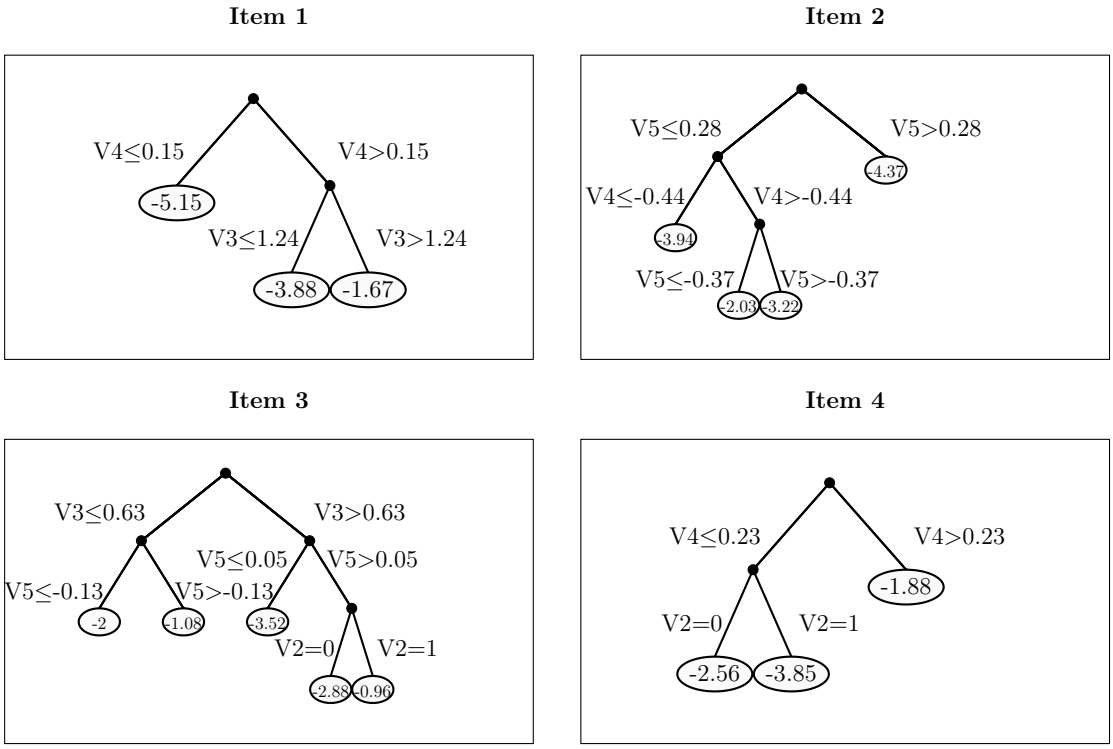

Figure 5.1.: Example of item focussed trees from simulation scenario 1 (logistic DIFtree strong DIF setting)

Then, the stability of the item parameters is assessed with respect to each covariate. This is done by calculating the deviations from the joint model for every person. The deviations are ordered according to every covariate (for example for covariate age from the youngest test taking person to the oldest). If the ordering exhibits a systematic change in the range of possible values of the covariate instead of a random fluctuation, DIF is said to occur. The sample is split according to the covariate with the greatest deviations, if they are found to be significant by means of a generalized M-fluctuation test (for details, see Strobl et al. (2015)). The cutoff point is chosen such that it leads to the highest improvement of the model fit (equivalent to the greatest reduction of the likelihood). This procedure is applied recursively in the resulting subsamples until no more significant changes in the deviations are found.
Since the main focus of the methods presented in this thesis is to detect DIF on the item level, Rasch trees are not included in the simulations. But they are computed in the empirical example and their findings are briefly compared to the other methods in chapter 7.

5.3. DIFtree

Item focussed trees (IFT), introduced by Tutz and Berger (2016), combine the flexibility aspect of decision trees (regarding the definition of subgroups) with the requirement of detecting DIF on the item level. They can deal with uniform and non-uniform DIF. In

the following, in coincidence with the other methods, the focus is on uniform DIF only. Two different versions of item focussed trees/*DIFtree* exist: the first one is based on the DIF model (called item focussed Rasch trees or IFRT in the following) and the second one combines trees with the logistic regression methodology (called item focussed logistic trees or IFLT in the following). In general, one global model (the Rasch model or a logistic model, depending on the version of *DIFtree*) is built and in order to account for uniform DIF, the item difficulty parameter is partitioned, such that there is one item difficulty parameter for each subgroup in the resulting tree for the respective item.

5.3.1. Item focussed Rasch trees

Same as *DIFlasso* and *DIFboost*, item focussed Rasch trees are based on the extended Rasch model (see (3.1)). Due to the nature of tree-based procedures, the model is sequentially growing as the trees grow, including an additional binary split in every step. In the end, the predictor includes a coefficient for every region of the predictor space, that is distinguished by the terminal nodes of the trees.

Concept

First, a base model, the ordinary Rasch model with predictor $\eta_{pi} = \theta_p - \beta_i$, is estimated. Then, the item difficulty part is recursively partitioned yielding varying item difficulties for the different partitions of the feature space. The predictor after the first split, for a metrically or ordinally scaled variable j according to split point c_j, can be denoted as (see Tutz and Berger (2016))

$$\eta_{pi} = \theta_p - [\gamma_{il}^{[1]} I(x_{pj} \leq c_j) + \gamma_{ir}^{[1]} I(x_{pj} > c_j)] \; . \tag{5.1}$$

I(.) describes the indicator function, that is equal to one if the corresponding expression is true and zero otherwise. It is used to choose the respective item difficulty $\gamma_{il}^{[1]}$ or $\gamma_{ir}^{[1]}$ for every person depending on the group membership. In this case here, person p is assigned to the left node if its value of the predictor variable j, x_j, is below or equal to the threshold value c_j and to the right node if the value is above the threshold. Accordingly, the predictor for a person in the left node would be $\eta_{pi} = \theta_p - \gamma_{il}^{[1]}$ and $\eta_{pi} = \theta_p - \gamma_{ir}^{[1]}$ otherwise.
Here, the differences to *DIFlasso* or *DIFboost* become apparent: the item difficulty for metric covariates is not directly composed from the value of the predictor variable of person p and item difficulty parameter γ, but all persons on the same side of the threshold value are related to the same item difficulty. For categorical variables, persons will be grouped into two categories since only binary splits are allowed. Of course, the same variable could be split further in a subsequent step of the procedure.
If another split for item i is performed, be it in the proximate or in a later iteration, the resulted left or right node of the first split would be further partitioned. For example, the right node specified by $I(x_{pj} > c_j)$ could be further split, this time according to categorical variable s, where S is a subset containing one or more categories of variable s and $\bar{S}$ is the complement. This yields daughter nodes $I(x_{pj} > c_j)I(x_{ps} \in S)$ and $I(x_{pj} > c_j)I(x_{ps} \in \bar{S})$

and the linear predictor is given by

$$\eta_{pi} = \theta_p - [\gamma_{il}^{[1]} I(x_{pj} \leq c_j) + \gamma_{il}^{[2]} I(x_{pj} > c_j) I(x_{ps} \in S) + \gamma_{ir}^{[2]} I(x_{pj} > c_j) I(x_{ps} \in \bar{S})] \ , \quad (5.2)$$

where $\gamma_{il}^{[2]}$, $\gamma_{ir}^{[2]}$ are the weights on the new split, that replace $\gamma_{ir}^{[1]}$ and define the new item difficulties in the subregions of $I(x_{pj} > c_j)$.
In every step of the procedure, every possible combination of item, variable and split point is tested and the combination is chosen that is found to be most significant according to a concept that uses maximal value statistics and a permutation test. Details of the procedure, that determines whether to perform further splitting at all, and if so, according to which combination of item, variable and split point, will be given in the next paragraph.
To facilitate the representation only metric or ordinal variables are considered in the following with a certain split point c_j. In general, each node can be represented by a product of several indicator functions:

$$node(\mathbf{x}_p) = \prod_{b=1}^{B} I(x_{pj_b} \leq c_{j_b})^{a_b} + I(x_{pj_b} > c_{j_b})^{1-a_b} \quad (5.3)$$

where B is the number of indicator functions or branches of the tree, c_{j_b} describes the selected cutoff point in variable j_b and $a_b \in \{0, 1\}$ indicates, which of the indicator functions, above or below threshold, is involved. Then, the final predictor for person p and item i with terminal nodes $l = 1, ..., L_i$ can be denoted as:

$$\eta_{pi} = \theta_p + \sum_{l=1}^{L_i} \gamma_{il} \ node_{il}(\mathbf{x}_p) = \theta_p + tr_i(\mathbf{x}_p) \quad (5.4)$$

γ_{il} denotes the item difficulties in the terminal nodes. Let $\sum_{l=1}^{L_i} \gamma_{il} \ node_{il}(\mathbf{x}_p) = tr_i(\mathbf{x}_p)$, then $tr_i(\mathbf{x}_p)$ takes the values of the respective item difficulties if a tree is built for item i. If no tree is built for item i, $tr_i(\mathbf{x}_p)$ equals β_i, the item parameter of the simple Rasch model over all persons p.

Splitting procedure

The iterative process of growing trees is mainly determined by the decision whether the feature space should be further partitioned or not and, if so, according to which combination of item, variable and split point.
For both, item focused Rasch and item focused logistic trees, the same test-based concept is used. The test statistic used here is the likelihood ratio test statistic. In every recursive partitioning step, the value of the test statistic is obtained for every possible combination of item, variable and split point. The corresponding null hypothesis is: $H_0 : \gamma_{il} = \gamma_{ir}$. Note, that this is impossible for metric variables and therefore, 20 quantiles are used and tested as possible split points. For every item and variable j the maximal value statistic $T_j = \max_{c_j} T_{j \ c_j}$ over all possible split points is computed. Then, a permutation test is carried out for the combination of item and variable that has the largest T_j. A permutation test is based on the intuition that if there are no group differences (resulting in the decision that no further splitting is done) the value of the test statistic should be about the same

as the test statistic after a random permutation of the persons' group memberships under which the statistic can be computed. Since the number of possible permutations gets very large, usually a fixed number of permutations, say 1000, is computed. This results in a distribution for the test statistic T_j based on the different values of T_j from the permutations. It is possible to obtain a p-value from the sample-specific permutation distribution, that is the number of increasingly sorted permuted T_j's above the observed T_j without permutation divided by number of permutations. In order to account for the number of covariates, the significance level used here is the overall significance level alpha divided by the number of covariates. If the p-value for the permutation test is significant, DIF is said to occur for the respective item and variable. Further splitting is performed according to the split point c_j for which $T_{j\ c_j}$, the test statistic for item i and variable j over all possible split points c_j, had the smallest p-value.
This proceeding is repeated until no more splits are found to be significant in the permutation test.

Algorithm IFRT

Summarizing the previous subsections, the *DIFtree* algorithm for the detection of uniform DIF and underlying Rasch model is defined as:

DIFtree (Rasch, Uniform DIF)

Step 1 (Initialization)
Set counter $\nu = 1$

- For all item $i = 1, ..., I$ fit all the candidate Rasch models with predictor

$$\eta_{pi} = \theta_p + [\gamma_{il}\ I(x_{pj} \leq c_{ijk}) + \gamma_{ir}\ I(x_{pj} > c_{ijk})],$$
$$j = 1, ..., m,\ \ k = 1, ..., K_j$$

- Select the model that has the best fit. Let c_{i_1,j_1,k_1} denote the best split which is found for item i_1 and variable x_{j_1}
- Select the item and variable with the largest value of T_j. Carry out permutation test for this combination with significance level α/m. If significant, fit the selected model yielding estimates $\hat{\beta}_i, \hat{\gamma}_{i_1}, \hat{\gamma}_{i_2}$ and nodes $node_{i_1}, node_{i_2}$, set $\nu = 2$. If not significant, stop (meaning no DIF detected)

Step 2 (Iteration)

- For all items $i = 1, ..., I$ and already built nodes $l = 1, ..., L_{iv}$, fit all the candidate DIF models with new intercepts

$$\gamma_{i,L_{iv}+1}\ node_{il}\ I(x_{pj} \leq c_{ijk}) + \gamma_{i,L_{iv}+2}\ node_{il}\ I(x_{pj} > c_{ijk})$$

  for all j and possible remaining split points c_{ijk}.

- Select the model that has the best fit, yielding split point c_{i_ν,j_ν,k_ν} which is found for item i_ν and variable x_{j_ν} in node $node_{i_\nu,l_\nu}$

- Select the item and variable with the largest value of T_j. Carry out permutation test for this combination with significance level α/m. If significant, fit the selected model yielding additional estimates $\hat{\gamma}_{i_\nu,L_{i_\nu}}, \hat{\gamma}_{i_\nu,L_{i_\nu}}$ and nodes $node_{i_1}, node_{i_2}$ and set $\nu = \nu + 1$.

Step 3 (Stop)

- Stop if permutation test is not significant

5.3.2. Item focussed logistic trees

A second version of item focussed trees uses the logistic model as underlying model for the *DIFtree* procedure.

Concept

The basic logistic model, as introduced in subsection 2.2.3 can be generalized to include multiple, possibly continuous or categorical predictor variables, that might induce DIF. It has the form (Berger and Tutz, 2016):

$$log\left(\frac{P(Y_{pi}=1|S_p,\mathbf{x}_p)}{P(Y_{pi}=0|S_p,\mathbf{x}_p)}\right) = \eta_{pi} = \beta_{0i} + S_p\beta_i + \mathbf{x}_p^T\boldsymbol{\gamma}_i \tag{5.5}$$

Again, β_{0i} is the intercept parameter, S_p denotes the test score of person p, and $\mathbf{x}_p^T\boldsymbol{\gamma}_i$ are vectors including the person-specific covariate values and the DIF-parameters γ for item i.

The logistic model in the tree framework

If no split is found for item i and no tree is built, the predictor of model (5.5) reduces to $\eta_{pi} = \beta_{0i} + S_p\beta_i$. If a split is found, the first split corresponding to the logistic model and according to a metric or categorical covariate x_j in tree notation is:

$$\eta_{pi} = S_p\beta_i + [\gamma_{il}^{[1]}I(x_{pj} \leq c_j) + \gamma_{ir}^{[1]}I(x_{pj} > c_j)] \tag{5.6}$$

with the indicator function being defined as above. For IFLT, if a split is carried out, the intercept parameter β_{0i} is not estimated as a separate parameter, but is contained in the item difficulty of the subgroups γ_{il}/γ_{ir}.
In order to determine whether splitting is accomplished or not one relies again on the concept of maximally selected statistics as described in the previous section.
Using node representation (5.3), the general predictor for item focussed logistic trees is:

$$\eta_{pi} = S_p\beta_i + \sum_{l=1}^{L_i}\gamma_{il}\ node_{il}(\mathbf{x}_p) = S_p\beta_i + tr_i(\mathbf{x}_p) \tag{5.7}$$

Algorithm IFLT

The *DIFtree* algorithm, using an underlying logistic model, can be summarized as:

DIFtree (Logistic, Uniform DIF)

Step 1 (Initialization)
Set counter $\nu = 1$

- For all item $i = 1, ..., I$ fit all the candidate logistic models with predictor

$$\eta_{pi} = S_p\beta_i + [\gamma_{il}\ I(x_{pj} \leq c_{ijk}) + \gamma_{ir}\ I(x_{pj} > c_{ijk})],$$
$$j = 1, ..., m,\ k = 1, ..., K_j$$

- Select the model that has the best fit. Let c_{i_1,j_1,k_1} denote the best split which is found for item i_1 and variable x_{j_1}
- Select the item and variable with the largest value of T_j. Carry out permutation test for this combination with significance level α/m. If significant, fit the selected model yielding estimates $\hat{\beta}_i, \hat{\gamma}_{i_1}, \hat{\gamma}_{i_2}$ and nodes $node_{i_1}, node_{i_2}$, set $\nu = 2$. If not significant, stop (meaning no DIF detected)

Step 2 (Iteration)

- For all items $i = 1, ..., I$ and already built nodes $l = 1, ..., L_{iv}$, fit all the candidate logistic models with new intercepts

$$\gamma_{i,L_{iv}+1}\ node_{il}\ I(x_{pj} \leq c_{ijk}) + \gamma_{i,L_{iv}+2}\ node_{il}\ I(x_{pj} > c_{ijk})$$

  for all j and possible remaining split points c_{ijk}.
- Select the model that has the best fit, yielding split point c_{i_ν,j_ν,k_ν} which is found for item i_ν and variable x_{j_ν} in node $node_{i_\nu,l_\nu}$
- Select the item and variable with the largest value of T_j. Carry out permutation test for this combination with significance level α/m. If significant, fit the selected model yielding additional estimates $\hat{\gamma}_{i_\nu,L_{i_\nu}}, \hat{\gamma}_{i_\nu,L_{i_\nu}}$ and nodes $node_{i_1}, node_{i_2}$ and set $\nu = \nu + 1$.

Step 3 (Stop)

- Stop if permutation test is not significant

Logistic IFT for the detection of non-uniform DIF

One advantage of item focussed trees for the detection of differential item functioning is their flexibility. In addition to uniform DIF, logistic IFTs can also detect non-uniform DIF, which, for example, *DIFlasso* and *DIFboost* are not capable of. This is realized by incorporating group specific slopes in the logistic model, for more information, see Berger and Tutz (2016).

6. Simulations

In the following, *DIFlasso*, *DIFboost* and *DIFtree* should be compared by means of a simulation study. This chapter consists of three parts: in section 6.1, the general settings are listed, that hold for both simulation scenarios as well as the considered criteria to evaluate the performance of the different methods. Section 6.2 covers the first simulation scenario, where data is generated according to the DIF model (3.1). In the second scenario, that is discussed in section 6.3, the data is again generated according to the DIF model but the underlying DIF structure is different.
All the analyses presented in this thesis were conducted with the R software (R Core Team, 2015), together with the packages "DIFlasso" (version 1.0-2), "DIFboost" (version 0.1) and "DIFtree" (version 2.0.4).

6.1. Settings

Parameters

To ensure comparability, as many parameters as possible are kept consistent over both settings. These are the number of observations P = 500, the number of items I = 20 and the number of DIF items $\#I_{DIF} = 4$. Accordingly, 16 Items are DIF-free: $\#I_{NO-DIF} = 16$. In both settings, five covariates are considered (m=5), of which the first two ones are binary distributed: $x_1 = x_2 \sim B(0.5)$ and the other three are metrically distributed, drawn from a standard normal distribution: $x_3 = x_4 = x_5 \sim N(0, 1)$. This is equal to working with standardized person characteristics, where the variance of the components is one. Both, person and item parameter are drawn from a standard normal distribution as well. $\theta_p \sim N(0, 1)$ and $\beta_i \sim N(0, 1)$.
In both scenarios, the data is generated according to the DIF model, but the way how the covariates account for DIF varies between the scenarios and is explained in the respective sections. Each scenario is repeated 100 times.

Performance criteria

The performance of the methods will be evaluated using several different criteria, including mean squared errors (mse's) and detection rates.
In order to evaluate how well the person and item parameters are estimated (in terms of how close they are to the true parameters), the respective mse's are calculated. The mse of the person parameter θ_p is defined as the squared difference between the estimated and the true person parameters averaged over all persons: $MSE_\theta = \sum_p (\theta_p - \hat{\theta}_p)^2 / P$. The mse of the group-specific item difficulty is calculated by $MSE_{\beta\gamma} = \sum_p \sum_i [(\beta_i + \mathbf{x}_p^T \boldsymbol{\gamma}_i) - (\hat{\beta}_i + \mathbf{x}_p^T \hat{\boldsymbol{\gamma}}_i)]^2 / (I \cdot P)$ and denotes the squared difference between the estimated and the true person-specific item difficulty averaged over all persons and items. Note that for the *DIFtree* procedure with underlying logistic model the mse's can not be calculated.

In addition to the mse's, true and false positive rates are calculated:

- The true positive rate on the item level

  $$TPR_I = \frac{1}{\#I_{DIF}} \sum_{i:\delta_i \neq 0} I(\hat{\delta}_i \neq 0)$$

  describes the percentage of how many of the DIF items are correctly identified as DIF items.

- The false positive rate on the item level

  $$FPR_I = \frac{1}{\#I_{NO-DIF}} \sum_{i:\delta_i = 0} I(\hat{\delta}_i \neq 0)$$

  denotes the percentage of how many of the DIF-free items are falsely diagnosed as DIF items.

- In addition, except for the *DIFlasso* procedure with group lasso penalty, the true and false positive rates can also be calculated on the level of each item-variable combination. The true positive rate on the level of each item-variable combination is:

  $$TPR_{IV} = \frac{1}{\#I_{DIF-VAR}} \sum_{i,j:\delta_{i,j} \neq 0} I(\hat{\delta}_{i,j} \neq 0)$$

  with $\#I_{DIF-VAR} = \#I_{DIF} \cdot m_{I_{DIF}}$ and $m_{I_{DIF}}$ the number of DIF-inducing covariates per item

- The false positive rate on the level of each item-variable combination is denoted as:

  $$FPR_{IV} = \frac{1}{\#I_{NO-DIF-VAR}} \sum_{i,j:\delta_{i,j} = 0} I(\hat{\delta}_{i,j} \neq 0)$$

  with $\#I_{NO-DIF-VAR} = \#I_{NO-DIF} \cdot m + \#I_{DIF} \cdot m_{I_{NO-DIF}}$

6.2. Scenario 1

Data generation

In both scenarios, three different strength of DIF are considered (strong, medium, weak). The strength of DIF is measured by:

$$\frac{1}{\#I_{DIF}} \sum_{i=1}^{I_{DIF}} \left(\frac{1}{m} \sqrt{\sum_{j=1}^{m} \gamma_{ij}^2} \right) \tag{6.1}$$

For independent components, the variance of the person-specific item difficulties $\beta_i + x_p^T \gamma_i$, $V_i = \text{var}(\beta_i + x_p^T \gamma_i)$, takes the simple form $V_i = \sum_j \gamma_{ij}^2$. Standardized by the number of covariates m and averaged over all DIF items, this gives a measure of the DIF strength in the DIF items. The parameters of the γ-vectors are chosen such that (6.1) equals 0.25 in the strong DIF setting. A DIF strength of 0.25 corresponds to a value of 0.05, if $\frac{1}{m}\sqrt{V_i}$ is averaged over all items.

For every item, DIF is induced by three covariates. The chosen γ-vectors for the four DIF items in the strong DIF setting are $\gamma_1 = (-0.5, 0, 0.8, 0.8, 0), \gamma_2 = (0, 0.9, 0, 0.7, -0.8)$, $\gamma_3 = (0, 0.8, -0.6, 0, 0.5), \gamma_4 = (0.7, -0.7, 0, 0.8, 0)$. The γ-vectors for all other items equal zero.

For the medium DIF setting, the strong γ-parameters are multiplied by 0.75, resulting in a value of 0.1875 for the DIF strength (6.1). For the weak DIF setting, the parameters are multiplied by 0.5 respectively.

For each of the methods, some input parameters have to be fixed: For the *DIFlasso* procedure, both the group lasso penalty, as proposed by Tutz and Schauberger (2015), is used (setting *grouped*=TRUE) and the lasso penalty (3.3). For simplicity the two methods are referred to as the grouped and the ungrouped *DIFlasso* in the following. For the grouped *DIFlasso*, the number of different penalization parameters λ that are used during the procedure is set to 30 (*l.lambda*=30) and the degrees of freedom of the BIC are calculated according to Yuan and Lin (2006) (*df*="YL"). For the ungrouped *DIFlasso* *l.lambda*=100, because each group-specific parameter is treated independently during the penalization. For *DIFtree*, trees are built with the DIF model as underlying model (*model*="Rasch") and with the logistic model (*model*="logistic"), referred to as *DIFtree* Rasch and *DIFtree* logistic in the following. The global level of significance for the permutation test is set to *alpha*=0.05 and the number of performed permutation tests *nperm*=1000. Using *DIFtree* Rasch, a small Ridge penalty is applied, ensuring the existence of all model parameters (*penalized*=TRUE). For *DIFtree* logistic, the type of DIF has to be specified, which is uniform DIF here (*type*="udif"). Together with *DIFboost*, this leads to five different methods according to which DIF detection was performed. For the *DIFboost* procedure, the number of boosting iterations maximally performed in one iteration of the stability selection, is set to 1000 (*mstop*=1000), each model parameter has to be chosen in at least 90% (*cutoff*=0.9) of the 500 stability selection iterations (B=500) to enter the final model and the boosting procedure is stopped for each stability selection iteration as soon as q=12 base learners are found.

Results of scenario 1

Figure 6.1 shows boxplots of the mean squared errors of the person parameter θ over all simulation runs for the different methods. For most of the replications, the mse's are small, with only a few outliers. Overall, the differences between the methods are small.

Figure 6.2 displays boxplots of the mse's of the person-specific item difficulty. The grouped *DIFlasso* procedure has a little less outliers. The mse's of the *DIFboost* procedure are slightly higher than those of the other methods, but again, differences are small. The mse's of the person-specific item difficulty vary more over the different strengths of DIF. The weaker the DIF the smaller the mse's.

Table 6.1 displays the true and false positive rates for the different methods and settings averaged over all 100 replications. In the strong DIF setting, all methods detect the DIF items correctly in all replications, except for the *DIFtree* Rasch procedure (TPR of 99.8%). The ungrouped *DIFlasso* has a higher average FPR than the other methods. For the medium setting, *DIFboost* has the highest average TPR rate and the ungrouped *DIFlasso* has the lowest with 96%. In the weak setting, *DIFboost* outperforms the other methods, having an average TPR of 89.5%. *DIFboost* is followed by *DIFtree* with moderate DIF rates of 73.5 and 75.2%. *DIFlasso* cannot compete with the other methods in the weak setting of scenario 1, regarding the true positive rates.

Setting 1		DIFlasso grouped	DIFlasso ungrouped	DIFboost	DIFtree Logistic	DIFtree Rasch
strong DIF	TPR	1.000	1.000	1.000	1.000	0.998
	FPR	0.022	0.116	0.037	0.067	0.026
medium DIF	TPR	0.978	0.960	0.992	0.982	0.975
	FPR	0.008	0.066	0.035	0.056	0.031
weak DIF	TPR	0.095	0.338	0.895	0.735	0.752
	FPR	0.000	0.006	0.031	0.048	0.034

Table 6.1.: True and false positive rates on the item level for setting 1

Figure 6.3 gives some more information about the true and false positive rates, showing boxplots over the 100 simulation replications. In the strong DIF setting, the *DIFtree* Rasch procedure detects only three out of four DIF items in one replication, leading to a true positive rate of 99.8%. In the medium DIF setting, *DIFboost* and *DIFtree* do detect at least 75% of the DIF items correctly, whereas the *DIFlasso* procedure has also lower detection rates (50%, 0%) for some replications. In the weak setting of scenario 1, one can see that *DIFboost* still classifies all DIF items correctly in at least half of the replications, whereas the median true positive rate for *DIFtree* is 75%. The differences between the FPR's of the two methods are small here.

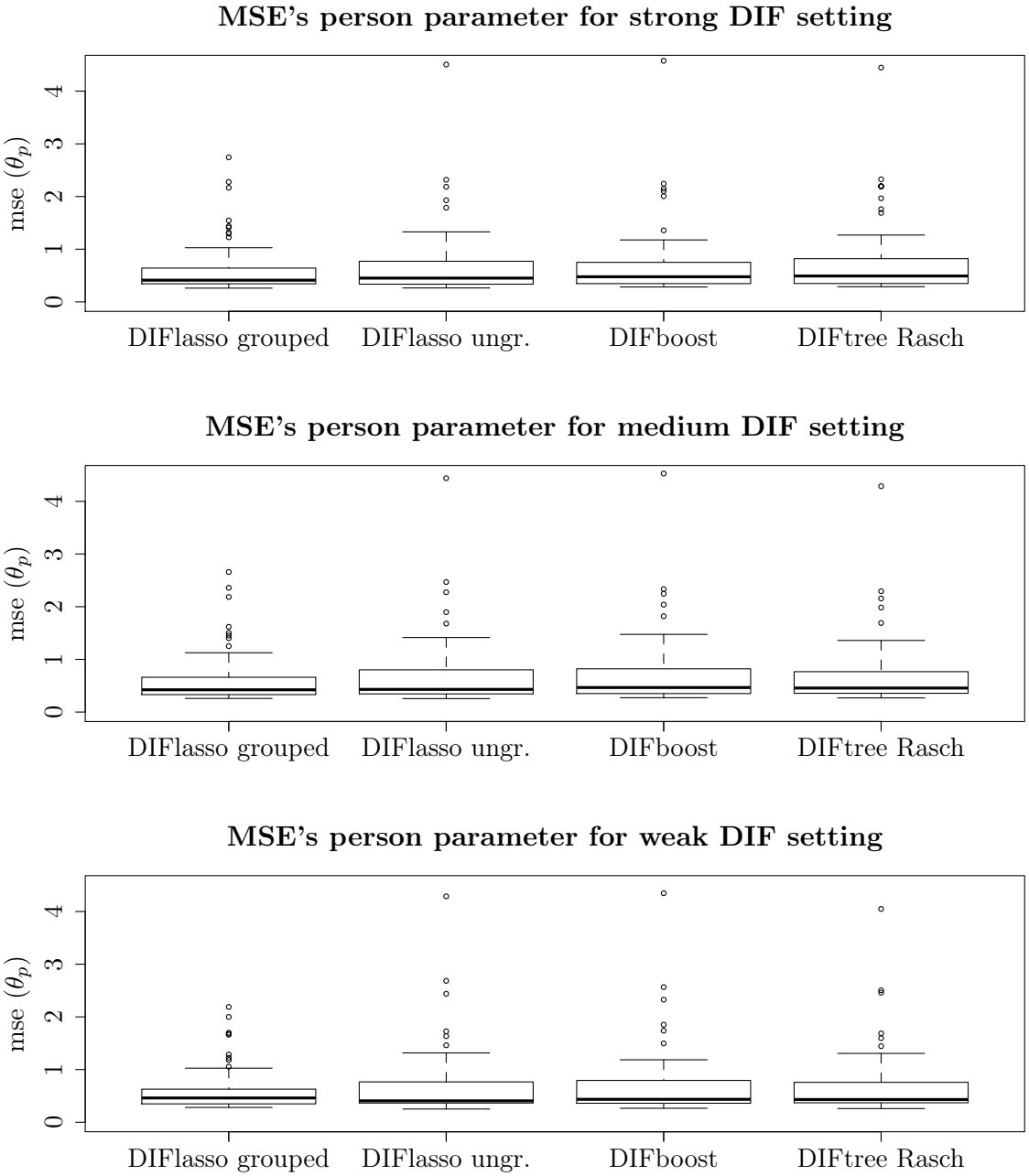

Figure 6.1.: Mean squared errors of the person parameter theta over all replications of scenario 1 for the different methods

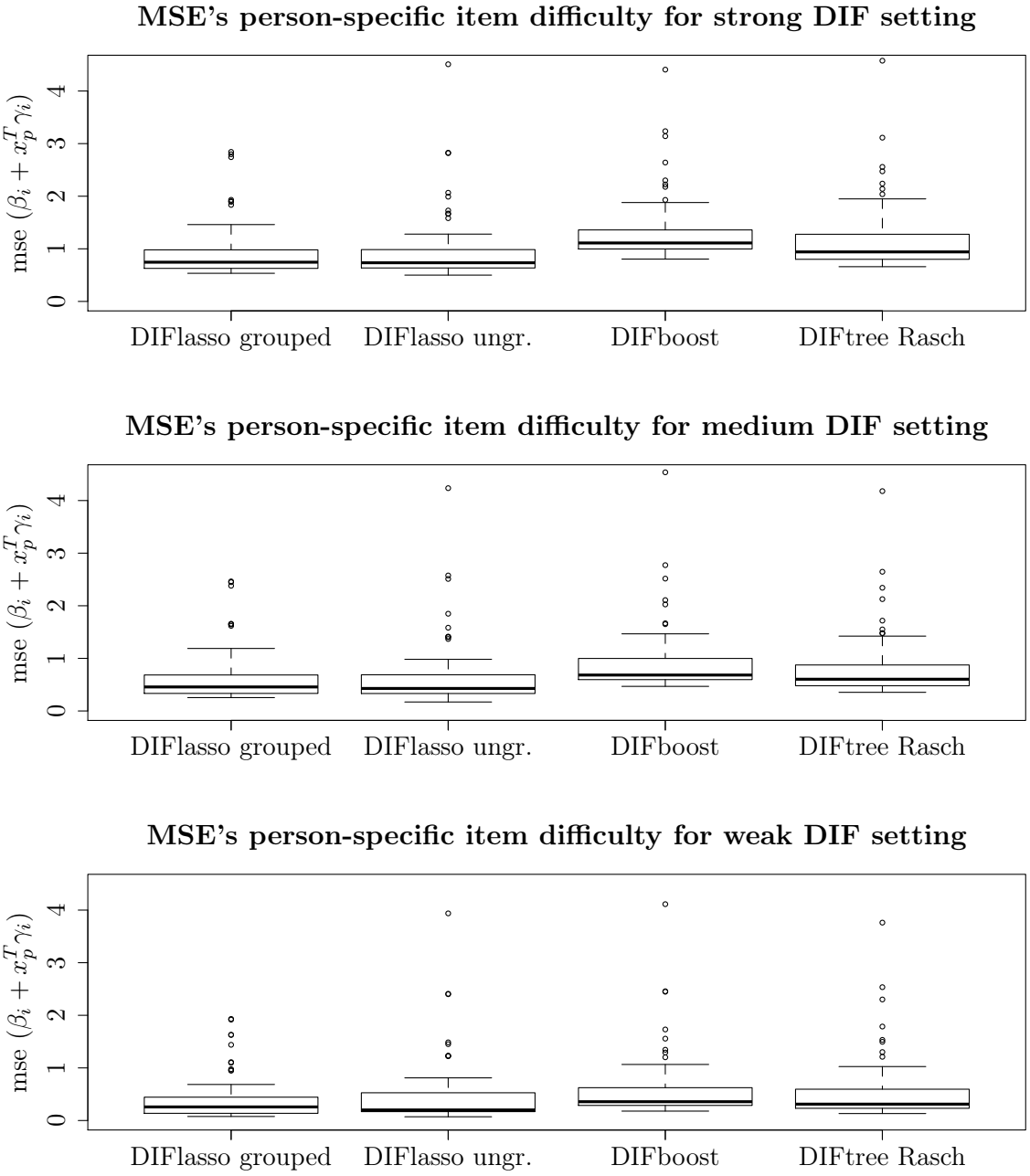

Figure 6.2.: Mean squared errors of the item parameter beta over all replications of scenario 1 for the different methods

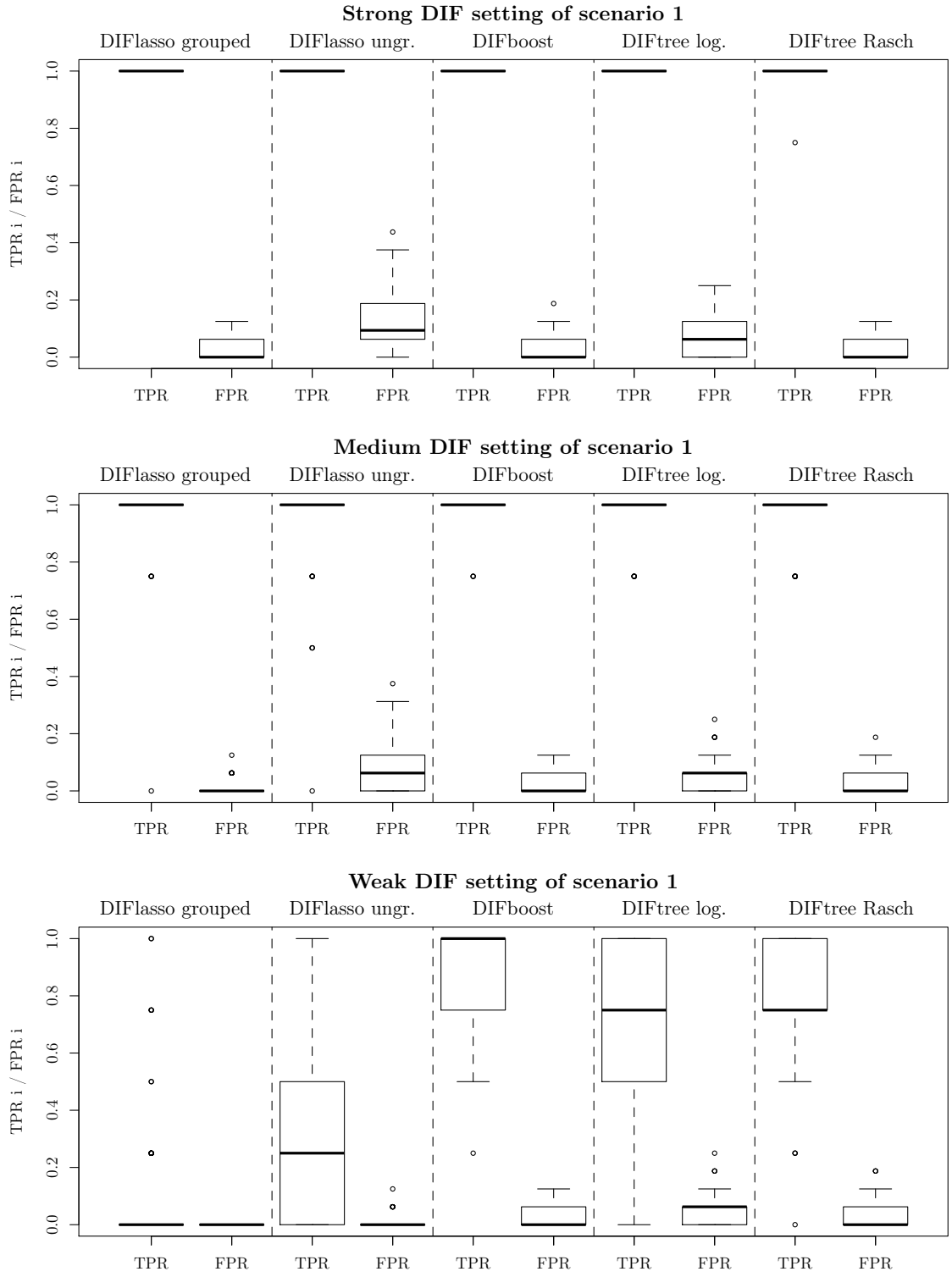

Figure 6.3.: True and false positive rates over all replications of scenario 1 for the different methods

Except for the grouped *DIFlasso* and the *DIFboost*, where either all or none of the parameters of a γ-vector for an item equal zero by definition, the TPR and FPR on the level of each combination of item and variable give additional information about how well the methods are able to not only identify DIF items but also to indicate the responsible variables on the item level:

Setting 1		DIFlasso ungrouped	DIFtree Logistic	DIFtree Rasch
strong DIF	TPR	0.887	0.688	0.671
	FPR	0.024	0.015	0.007
medium DIF	TPR	0.677	0.546	0.538
	FPR	0.013	0.012	0.007
weak DIF	TPR	0.134	0.310	0.317
	FPR	0.001	0.010	0.008

Table 6.2.: True and false positive rates for each item-variable combination for setting 1

For each item there are three DIF variables in scenario 1. In the strong and the medium DIF setting, the ungrouped *DIFlasso* has the highest TPR rates. In the weak setting, *DIFtree* slightly outperforms *DIFlasso* on the level of each item-variable combination, but overall, the detection rates are low.

6.3. Scenario 2

Data generation

Same as in scenario 1, three different strengths of DIF are considered (strong, medium, weak). Note that the DIF strength cannot be compared between the scenarios, meaning that strong DIF in the first scenarios is not necessarily the same as strong DIF in the second scenario.
In the tree framework, $V_i = \text{var}(\sum_l \gamma_{il}\ \text{node}_{il})$ describes the variance of the group-specific item parameters. Again, the average of V_i over the DIF items is used as a measure of the DIF strength. The DIF structure is taken according to Tutz and Berger (2016), slightly modified to incorporate one additional fifth covariate:

Item	DIF structure
1	$0.75c \cdot I(x1 = 1) + 0.75c \cdot I(x3 > 0.1)$
2	$-0.75c \cdot I(x1 = 1) - 0.75c \cdot I(x4 > 0.1)$
3	$0.8c \cdot I(x2 = 1) + 0.8c \cdot I(x5 > -0.1)$
4	$-0.8c \cdot I(x3 > 0.1) - 0.8c \cdot I(x5 > -0.1)$

Table 6.3.: DIF structure of the strong setting of scenario 2

Parameter c regulates the strength of the DIF. For strong DIF in scenario 2, parameter c=1, leading to a DIF strength of 0.41. For medium DIF, c=0.75 (0.23) and for weak DIF, c=0.5 (0.10). In scenario 2, there are two DIF variables per item and the DIF structure follows a tree structure that represents interactions between the two DIF variables.
The input parameter for the different methods, as described in section 6.2, remain the same in scenario 2.

Results of scenario 2

The mse's of the person parameter do not vary much across the different methods, as can be see from figure 6.4. Again, the grouped *DIFlasso* has slightly lower mse rates than the other methods.

More variation can be seen regarding the group-specific item parameters in scenario 2. In the strong and medium setting, the *DIFlasso* procedure has the lowest mse's, followed by *DIFtree* and *DIFboost*. The mse's for *DIFtree* Rasch increase as the strength of DIF decreases. The detection rates for the weak setting of scenario 2 are low. Taking a closer look on the parameters, the mse is mostly increased, because the parameter estimates are less close to the true parameters for the DIF items. This can be caused by unfavourable splits, i.e at the margins of the range of a numerical variable for weak DIF on the one hand and on the other hand, the fact that DIF items are not found has more influence on the mse for the *DIFtree* Rasch procedure than for the other methods for weak DIF.

In the strong setting of scenario 2, *DIFtree* performs best, with an accuracy of 90% (logistic *DIFtree*) and 87% (Rasch *DIFtree*), followed by *DIFboost*. DIFlasso does not perform very well in scenario 2. In the medium and weak DIF settings, *DIFboost* slightly outperforms *DIFtree*, which is a little surprising. In comparison to scenario 1, *DIFtree* holds the global significance level of 0.05, with an FPR of at most 0.044.

Setting 2		DIFlasso grouped	DIFlasso ungrouped	DIFboost	DIFtree Logistic	DIFtree Rasch
strong DIF	TPR	0.128	0.458	0.820	0.898	0.868
	FPR	0.001	0.006	0.036	0.042	0.041
medium DIF	TPR	0.005	0.160	0.620	0.552	0.525
	FPR	0.000	0.004	0.036	0.031	0.026
weak DIF	TPR	0.002	0.070	0.345	0.248	0.228
	FPR	0.000	0.013	0.044	0.039	0.029

Table 6.4.: True and false positive rates on the item level for setting 2

In scenario 2, *DIFlasso* can not compete with the other methods. This also holds for the comparison on the level of each item-variable combination, as can be seen from table 6.5. For all strength of DIF, the logistic *DIFtree* procedure performs slightly better than *DIFtree* Rasch.

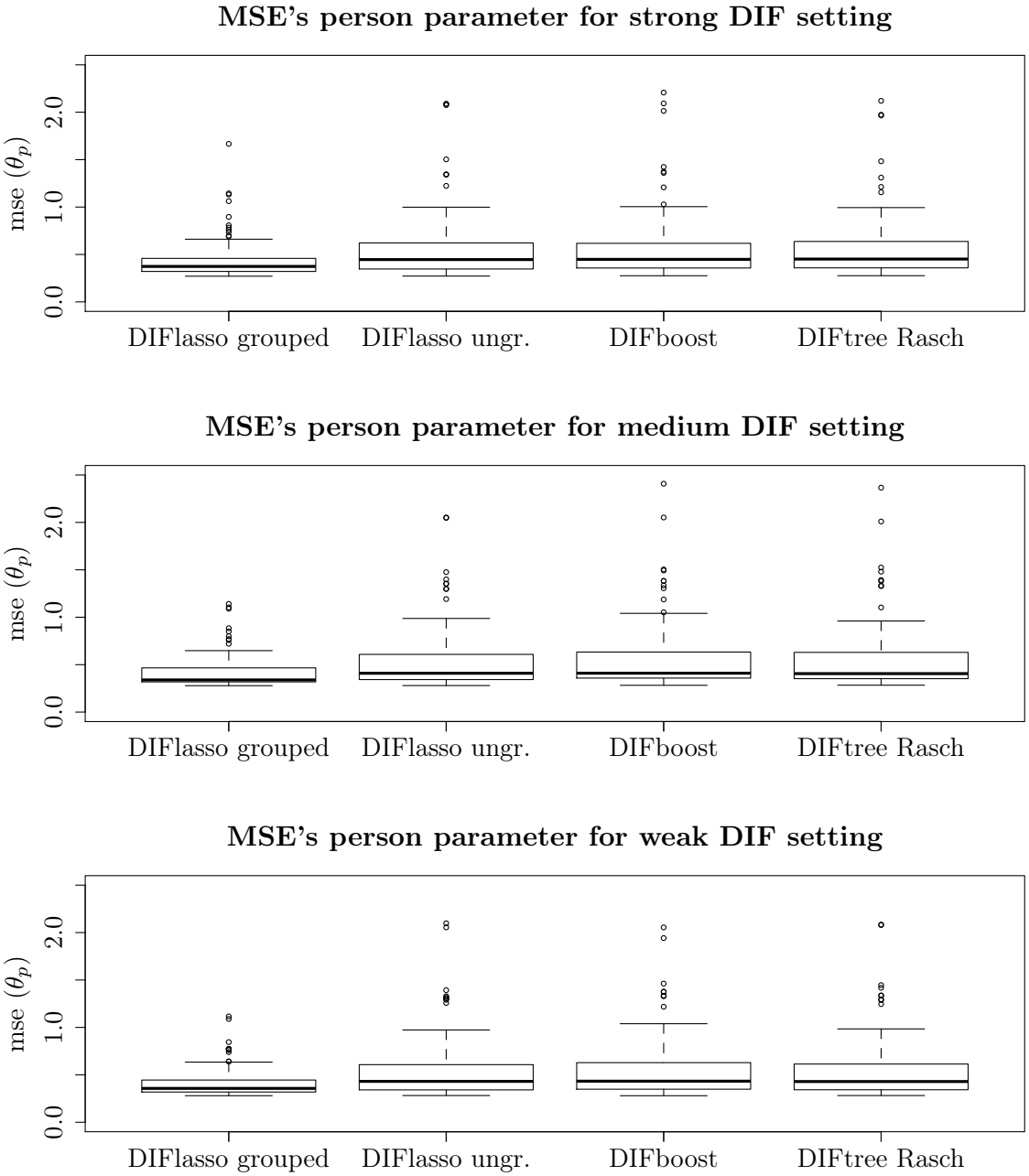

Figure 6.4.: Mean squared errors of the person parameter theta over all replications of scenario 2 for the different methods

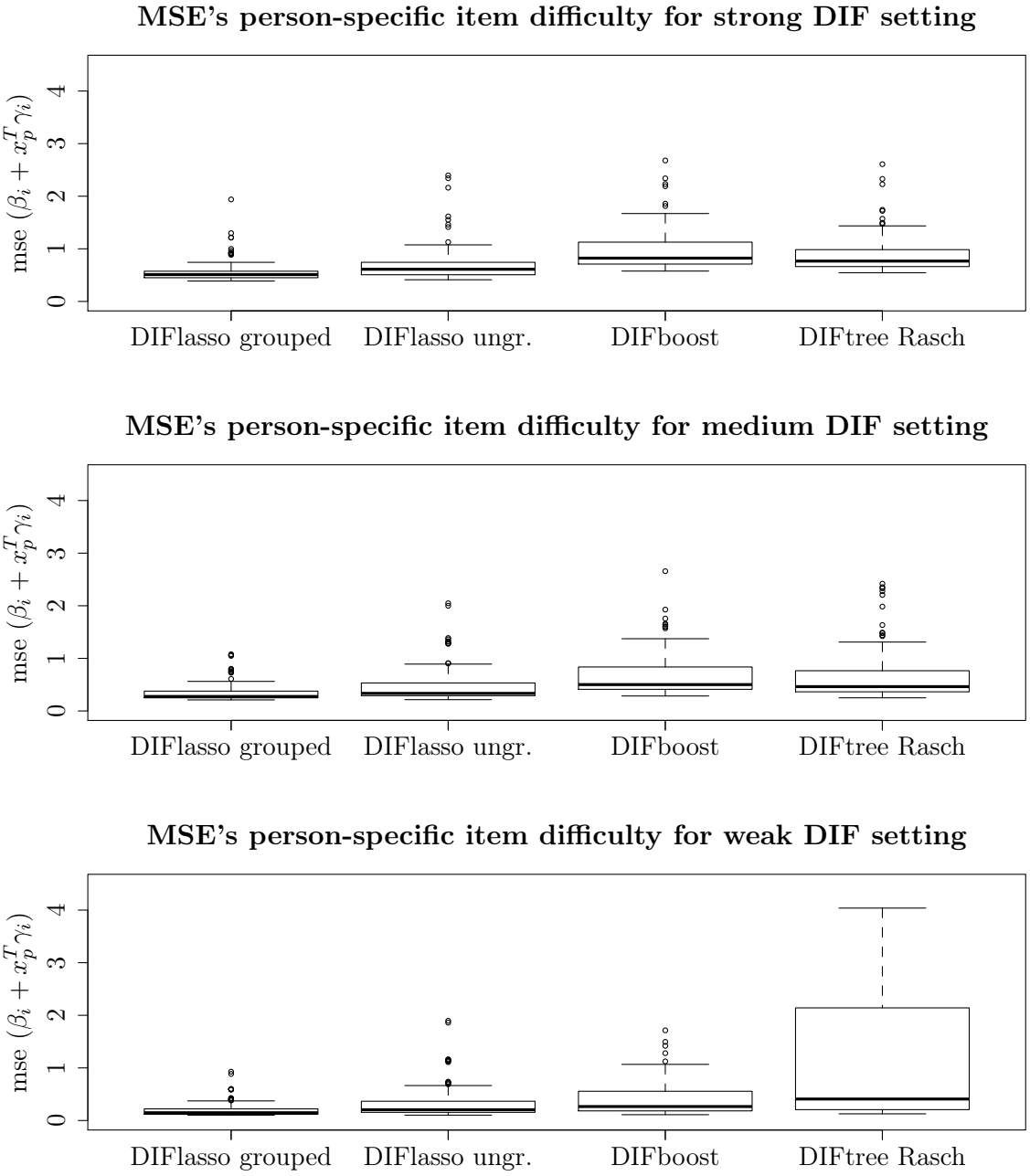

Figure 6.5.: Mean squared errors of the item parameter beta over all replications of scenario 2 for the different methods

Setting 2		DIFlasso ungrouped	DIFtree Logistic	DIFtree Rasch
strong DIF	TPR	0.311	0.602	0.560
	FPR	0.001	0.009	0.009
medium DIF	TPR	0.084	0.305	0.285
	FPR	0.001	0.007	0.006
weak DIF	TPR	0.030	0.118	0.112
	FPR	0.003	0.009	0.007

Table 6.5.: True and false positive rates for each item-variable combination for setting 2

6.4. Summary

Taking both scenarios into account, it seems that independent of the underlying DIF structure, *DIFboost* gives the best results in terms of how accurate DIF items are detected as DIF items. *DIFboost* is followed by *DIFtree*. *DIFtree* with underlying logistic model performs slightly better than with underlying Rasch model and the computation time is shorter. The larger the group differences the more the *DIFlasso* methodology can compete with the other methods. If the strength of DIF is considered to be weak, the true positives rates are much lower than for the other methods. For strong DIF, all methods give comparable results regarding true positive rates.
The mean squared errors of the person parameter do not vary much across the different methods and settings. The mean squared error of the person-specific item difficulty has some outliers when item focussed Rasch trees are used, especially when the group differences are small. This is when unfavourable splits might be found, i.e. at the margins of the range of the numerical variables and the respective parameter estimates are not reliable and lead to higher mse's. Also, DIF items not being detected as DIF items increases the mse's for *DIFtree* Rasch in some iterations of the weak setting of scenario 2.

7. Empirical example: Assessment of educational standards

In the following chapter, the practical behaviour of all methods should be compared by means of an empirical example using a data set from the Austrian "Bundesinstitut für Bildungsforschung, Innovation und Entwicklung" (Bifie). Among others, quality development, educational monitoring, conception of final examinations, applied educational research and information and consultancy services belong to the core tasks of Bifie. Educational monitoring includes the assessment of educational standards of all Austrian students in grade 4 (regarding areas of competence: German, Mathematics) and grade 8 (German, Mathematics, English). In a cycle of five years each area of competence is assessed once in a comprehensive survey. Here, data from the 8th grade assessment of mathematics standards from 2012 is analyzed (BIFIE, 2014).
In total, there are almost 80.000 8th grade students in Austria. A random sample of 851 surveyed students was provided. In addition to their performance on 48 test items, socio-economical background variables were captured. Therefore, both students and their parents answered additional questionnaires. For a detailed introductory documentation of the 8th grade assessment of mathematics standards, refer to Schreiner and Breit (2012). For a general technical documentation of the construction and design of standard tests, see Itzlinger-Bruneforth et al. (2016) and Kiefer et al. (2016). Kuhn and Kiefer (2013) refer to the test design of the standards assessment in mathematics more specifically.
The following section shortly describes the test design of the study, which is helpful for the understanding of how the sample is drawn exactly. In section 7.2, the predictor variables are explained and a descriptive overview of the data set is given. Section 7.3 covers the results of the DIF analysis for each of the methods. The chapter concludes with a short comparison of the empirical results.

7.1. Test design

The department of didactics in mathematics at the Alpen-Adria university Klagenfurt developed a model, that divides the concept of "mathematical competence" into three dimensions (see figure 7.1): a content area, an operational area and the level of complexity (Heugl et al., 2007). The content and operational dimensions have four different categories each, dimension complexity has three. For example, the categories for the operational dimension are: illustrating/model building, calculating, interpreting and arguing/justifying. Hence, the concept of "mathematical competence" is characterized by 4*4*3 = 48 different areas, described by tripels of the three dimensions.
This model of mathematical competencies is used for the design of the student's questionnaire for the assessment of mathematics standards. The test consists of 72 questions in total. The amount of questions is too large to be answered by a single student. Therefore, the items are divided into blocks. Every student answers a booklet of four out of six blocks and the blocks vary over students, such that in the end every item is solved by the same number of students. Every block was restricted to contain at least three items of each content area and of each operational area. In addition, at least four items of each complexity

level should be present in each block (Kuhn and Kiefer, 2013). Some of the methods cannot deal with missing values in the response matrix Y, thus one booklet ("Testheft 3") was randomly selected, from which the 851 observations were drawn.

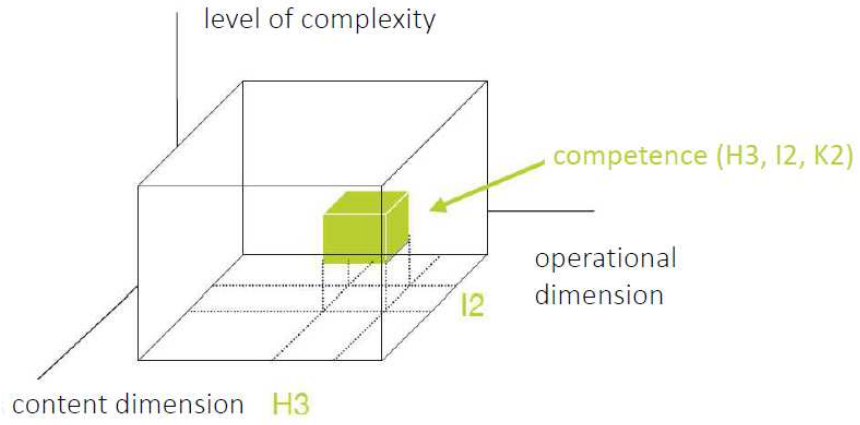

Figure 7.1.: A model of mathematical competencies (`https://www.bifie.at/node/49`)

7.2. Data description

In addition to 48 dichotomous variables that contain the test results for each of the students, the data set includes information on five socio-economical covariates:

- **female**: indication whether a student is female or male
- **language:** a three-categorical variable that shows whether the student's first language is German or not
- **migration:** also a three-categorical variable that expresses whether parents are born in Austria/Germany or not. Note that if the student's native country is Germany, this is not considered to be a migration background.
- **HISEI:** "Highest International Socio-Economic Index of occupational status". The "International Socio-Economic Index of occupational status" (ISEI) is calculated for both, father (FISEI) and mother (MISEI). The index takes the parents' occupation, education and salary into account. The salary itself is not retrieved directly from the parent's questionnaire, but derived from the 2008 International Standard Classification of Occupations. The HISEI is the highest parental ISEI.
- **sstat:** the social status is also not measured directly but derived from three covariates taken from both student's (SQ) and parent's questionnaire (PQ). It takes into account the number of books in the household (BOOK), parental education (PEDU) and HISEI and is calculated by:

$$\text{sstat} = \frac{1}{6}\Big(\text{HISEI}_{SQ} + \text{HISEI}_{PQ} + \text{BOOK}_{SQ} + \text{BOOK}_{PQ}\Big) + \frac{1}{3}\text{PEDU}_{PQ}$$

The three variables are z-standardized prior to calculation. Both, the social status and the HISEI were anonymized by rank swap before the data was made available. The correlation with the original variables is 0.98.

The covariates are coded as described in table 7.1.
DIFlasso and *DIFboost* require a dummy-matrix as function input. Therefore, the two multi-categorical variables migration and language were dummy-coded whereas the first category serves as the reference category.

variable name	variable values
female	0: male 1: female
language	1: first language of both parents is German 2: first language of either the father or the mother is German 3: both parents originally speak other languages than German
migration	1: inland, father and/or mother is born in Austria/Germany 2: second generation migrant (parents born abroad but child is born in Austria/Germany) 3: first generation migrant (parents and child born in a foreign country)
hisei	numerical values between 0 and 100
sozstat	numerical values between -2 and 2

Table 7.1.: Coding of the socio-economical predictor variables

Figure 7.2 shows the distribution of the test results and the five predictor variables in the provided data sample. The distribution of the number of correctly solved items per student approximately follows a normal distribution. Every student solved at least one item correctly and at most 47 (out of 48) items. Most students are Austrian natives (84%). A small amount of children is classified as a first generation migrant (12%) and 5% as a second generation migrant. The first language of 78% of the students is Austrian. When the parents are non-native speakers more often both parents are non-native speakers (15%) than just mother or father (7%). Variable sstat is approximately normally distributed. Note that all the displayed distributions and numbers refer to the sample data and do not necessarily, without further information, represent the test population.
There is a high correlation between the variables sstat and HISEI (Pearson's correlation coefficient of 0.78) and between the variables migration and language (Spearman rank correlation coefficient of 0.84). This should be kept in mind for the following DIF analysis, because if a variable is identified as a DIF variable, there is a high chance that there are also group differences for the correlated variable, even though it will not be assigned as a DIF variable.

All objects that have missing values in one or more of the covariates were excluded, assuming that the entries were missing at random. After removal, 773 of the 851 objects (91%) remain for further analysis. The covariates were standardized prior to handing them over to *DIFlasso* and *DIFboost*. In contrast, *DIFtree* requires un-standardized covariates.

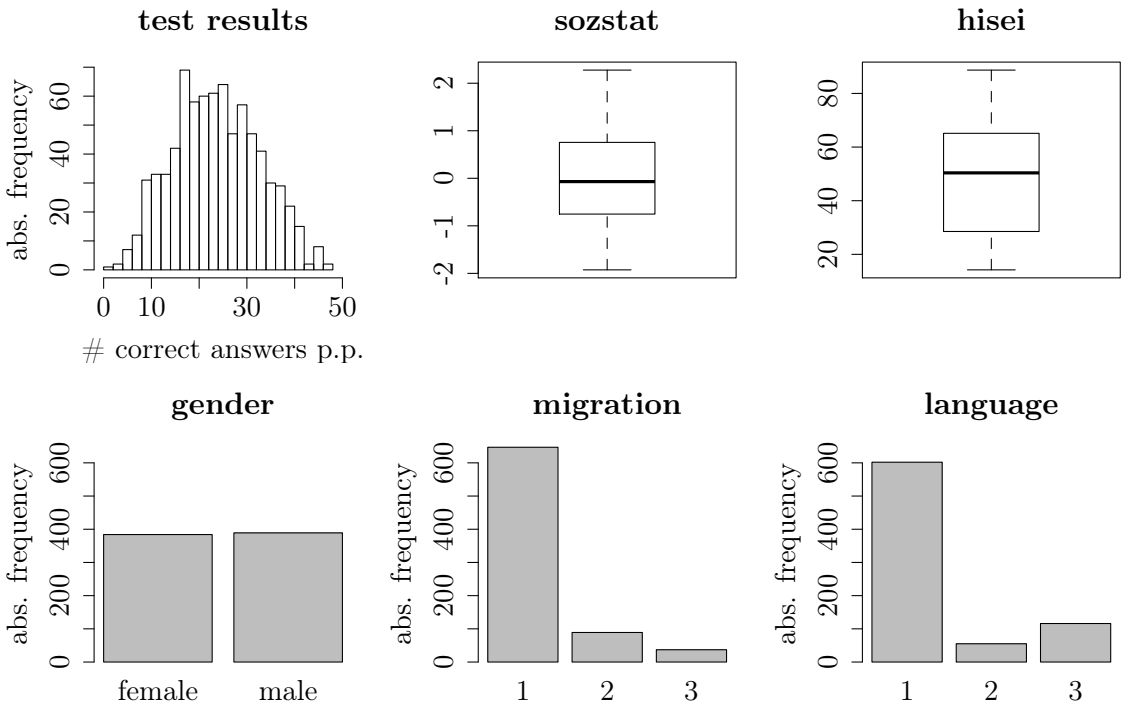

Figure 7.2.: Test results (upper left plot) and distribution of the five covariates

7.3. DIF analysis

In this section, the results of the DIF analysis are reported. Findings are displayed separately for every method.

Rasch trees

Rasch trees are useful to find DIF inducing variables on the global test level. Here, they are computed as a comparison to the other methods. However, they are less intuitive for the detection of DIF items in particular. For the Rasch tree analysis, the R-function *raschtree* is used, that is contained in the R-package *psychotree* (version 0.15-0) developed by Strobl et al. (2015).

The *raschtree* method finds two DIF-variables, gender and social status. Gender is the first splitting variable and social status the second splitting variable that comes into play when the gender of the test taking person is female (see figure 7.3). In the end, two big groups according to gender are formed and one very small group (n=30), containing female students with a low social status. In this group, the coefficients for four of the items (items 5, 8, 27, 45) can not be calculated, because none of the 30 students solved the item correctly. In subplot "Node 4" of figure 7.3, these four items are the ones with the lowest item parameters, but the value of -4.58 is somewhat misleading, since actually no parameter is estimated here.

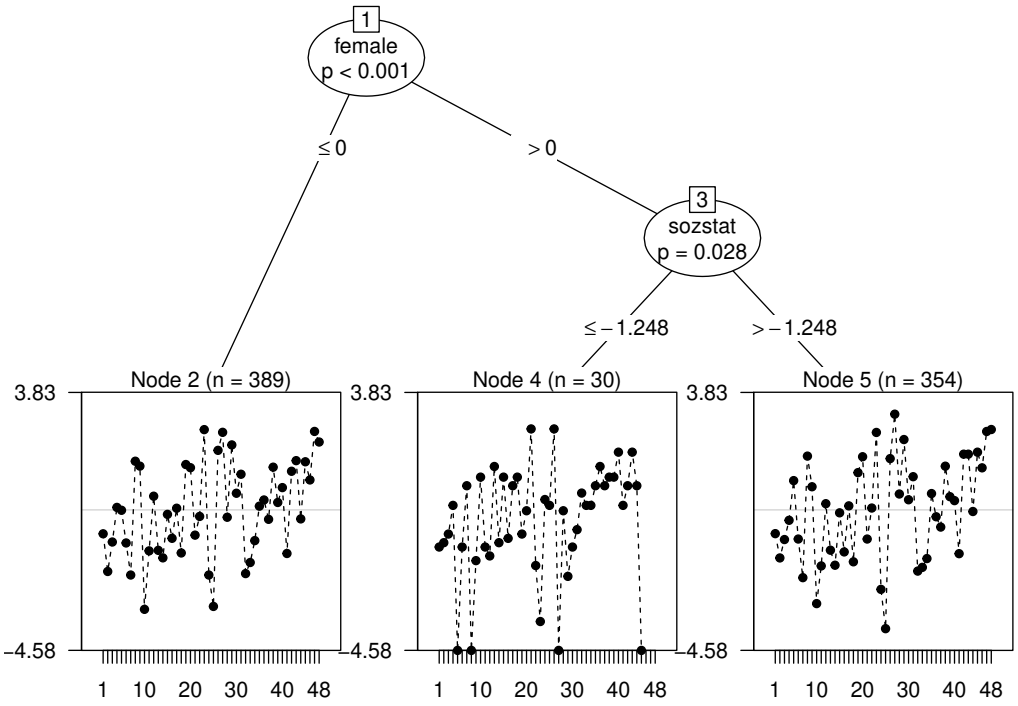

Figure 7.3.: Final estimated tree of the *raschtree* procedure

DIFlasso

For the *DIFlasso* procedure, four different settings were considered. Both the grouped and ungrouped lasso penalty were applied together with two different ways of calculating the degrees of freedom for the BIC, that determines the final model. The types of degrees of freedom are "Yuan-Lin" (YL) and "L2", see section 3.2.
Table 7.2 shows the number of DIF items that are found in the data set for each of the four different settings. Using the setting originally proposed by Tutz and Schauberger (2015) of a group lasso penalty and degrees of freedom according to Yuan and Lin (2006), no DIF items are found at all. For the grouped lasso/L2 setting, four DIF items are detected. For the ungrouped lasso/YL setting, one DIF item is found, which is item 5. In this context, it does not seem very plausible that the ungrouped lasso/L2 finds 20 DIF items.

DIFlasso setting	no. of DIF items	item	variable
grouped - df: YL	0	-	-
grouped - df: L2	4	5, 28, 38, 41	all
ungrouped - df: YL	1	5	gender
ungrouped - df: L2	20	3,5,...	...

Table 7.2.: DIF items found by the DIFlasso procedure under different settings

Figure 7.4 shows the findings for the ungrouped lasso/YL setting. The left plot of figure 7.4 visualizes the evolution of the γ-parameters for the different values of λ, that determine the strength of the penalization. The BIC-optimal model is indicated by the vertical dashed line. At this point, one DIF item (item 5) is detected in this setting. The right plot shows the group-specific coefficients of item 5. All coefficients are zero except for the gender, indicating that gender is the detected DIF variable. The group-specific coefficient for the gender is positive, meaning that the item is more difficult to solve for female students than for male students.

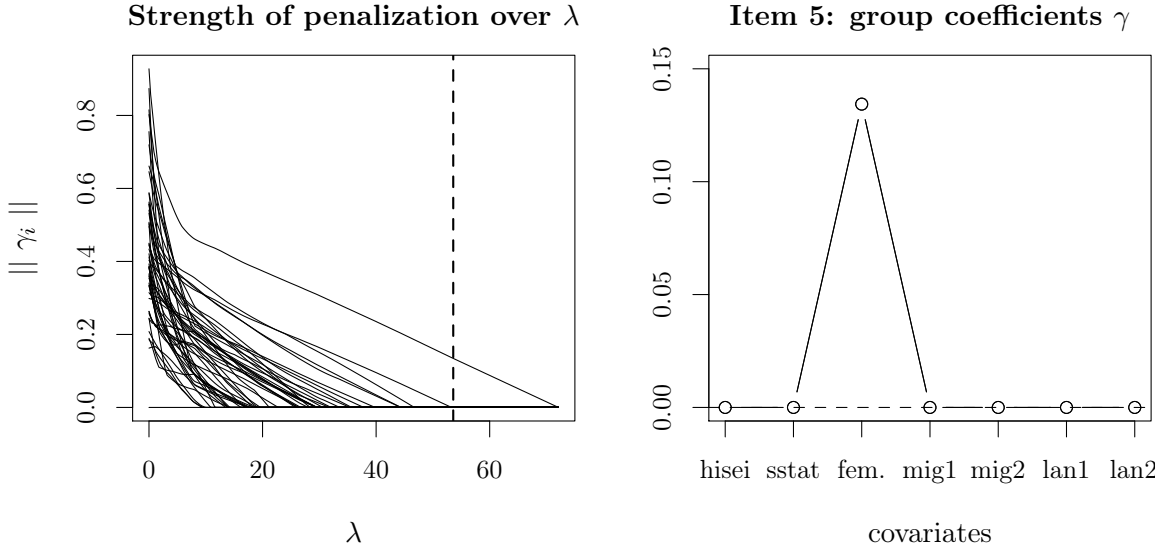

Figure 7.4.: Findings for the ungrouped lasso/YL setting: L2 norm of γ-coefficients vs. lambda (left) and γ-coefficients for the DIF item of the final model (right)

The grouped lasso/L2 setting detects four items as DIF items. The group-specific parameters are displayed in figure 7.5. Same as for the ungrouped lasso/YL setting, item 5 shows DIF and gender is again the variable with the largest group differences. Compared to item 5, the group differences for the other three items are relatively small. Since the variables were standardized prior to the analysis, the size of the group coefficients can be compared directly between variables. Item 28 is again more difficult to solve for female students. The higher the social status, the higher is the probability of a correct answer for item 41. The item difficulty is also higher if German is not the first language of the student.

DIFboost

For the *DIFboost* procedure it is important to keep parameter *mstop* sufficiently large to guarantee that enough base learners are found for every subsample of the boosting procedure. The larger the value of mstop, the higher is the time required for computing. Here, even if mstop is chosen to be very large, say 5000, not enough base learners are found according to parameter q, $q = 0.6 * I = 0.6 \cdot 48 \approx 29$. This is why q was set to 24. This means that at most 50% of the items can be classified as DIF items.
Using *DIFboost*, as developed by Schauberger and Tutz (2016), seven items (items 3, 5, 9, 12, 28, 38 and 41) are diagnosed as DIF items. For each item, all γ-parameters are

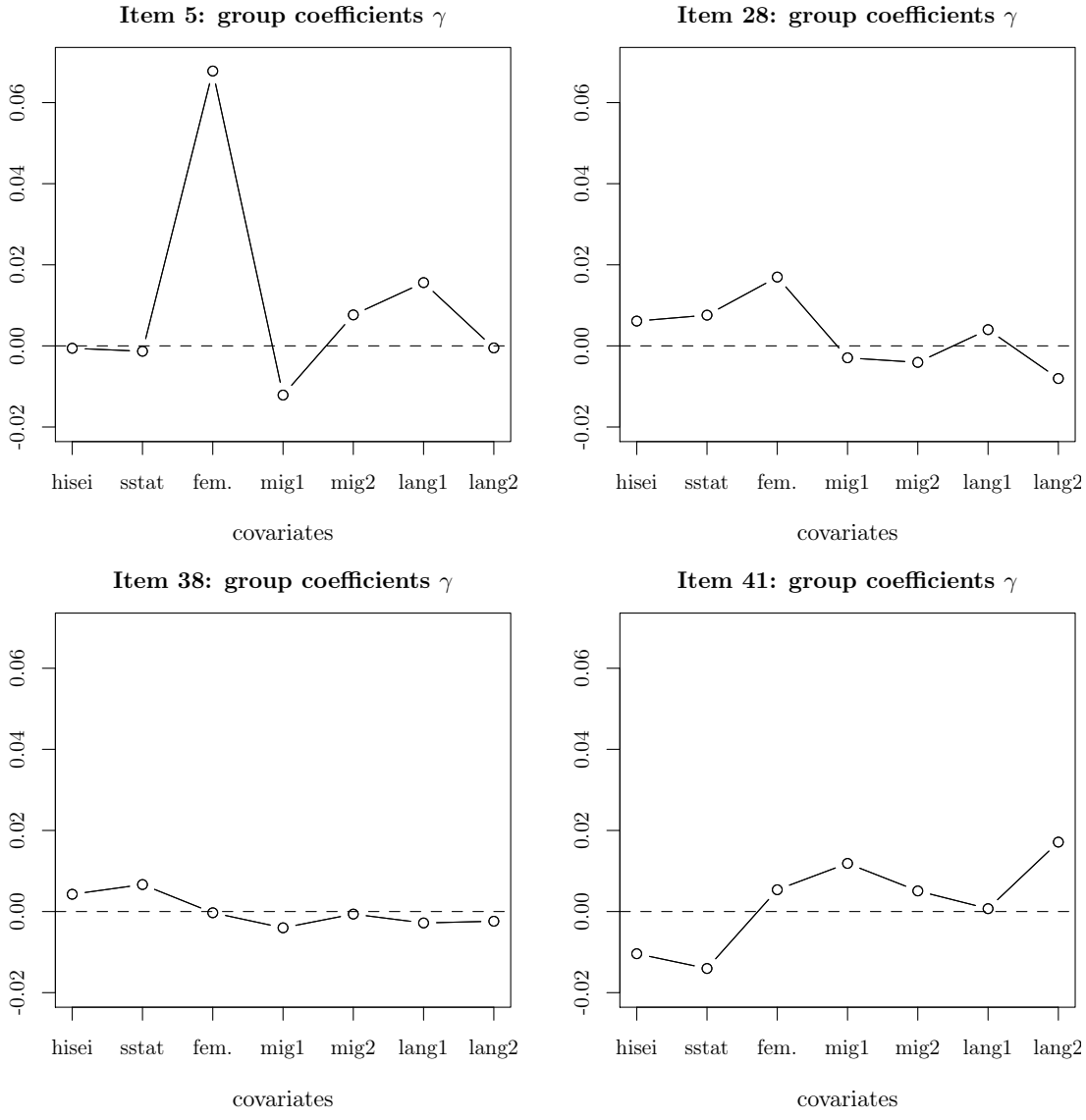

Figure 7.5.: Findings for the grouped lasso/L2 setting: γ-coefficients for the four DIF items of the final model

unequal to zero. Figure 7.6 displays the γ-coefficients for four of the seven detected items. These four items are chosen by other procedures as well and thus their classification as DIF items is regarded to be more reliable than if an item is chosen by one method only. Same as for *DIFlasso*, the *DIFboost* procedure detects item 5 (upper left plot) as a DIF item. The variable with the largest group-specific parameter is again the gender of the student. The direction conforms to the *DIFlasso* result as well. Item 9 is regarded as more difficult for male students and first generation migrants. Item 28 is again easier to solve for male students. For item 41, the results are similar to *DIFlasso* as well: the higher the social status, the lower is the item difficulty. In addition, the item is more difficult to solve for students whose first language is not German.

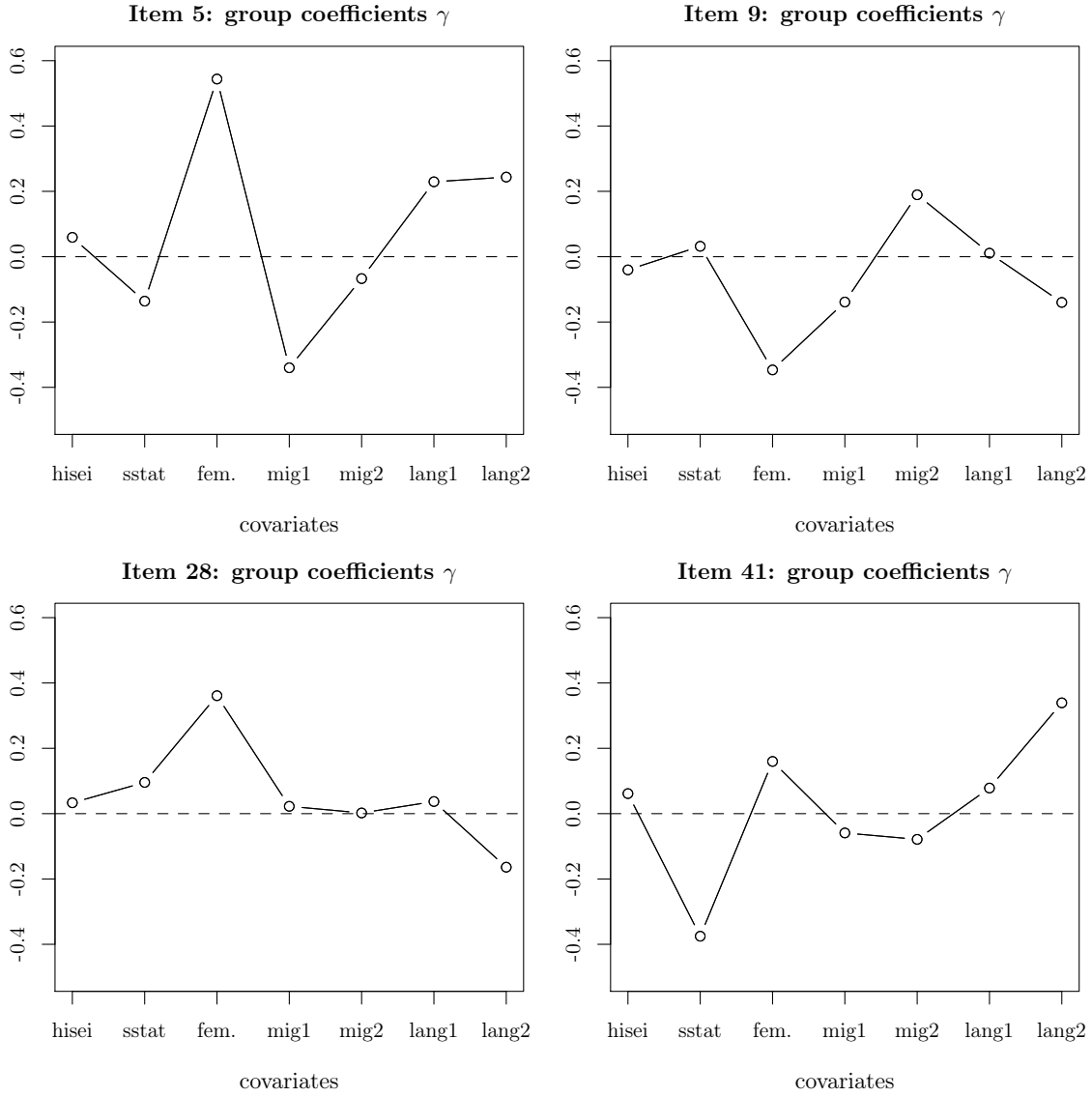

Figure 7.6.: Results of DIFboost: γ-coefficients of four DIF items detected by the DIFboost procedure

DIFtree

The *DIFtree* procedure with underlying Rasch model detects six DIF items in total. Applying the *DIFtree* algorithm with underlying logistic model, seven DIF items are found. Five of the items match for both settings and also the splitting variables are the same. These are items 5,9,28,31 and 41. Mostly, gender and social status are responsible for DIF. For each of the items, one DIF variable is found, except for item 41, where splitting is done according to the social status first and if the social status is below 1.44, another split is performed according to the migration background of the student. Figure 7.7 and 7.8 show the findings for the two *DIFtree* settings in more detail.

Three of the six items detected with item focussed Rasch trees split the observations according to the gender of the test taking student. Items 5 and 28 are easier to solve for male students, item 9 is more difficult. This matches with the *DIFlasso* and *DIFboost* results. The DIF variable of item 31 and 38 is the social status. In comparison to the other methods, where the relation between the test answer and the numerical variable social status is linear, the trees split the observations into two subgroups. The group with the higher social status has a higher probability of solving the item correctly. For item 41, the *DIFtree* Rasch procedure splits twice. The first split is according to the social status and the second split according to the migration background. However, the left node of the first split (students with a very low social status) contains only 40 of 773 students. The right node of the second split (students with a low social status and a migration background) contains only 18 observations, that mostly did not solve the item correctly. This decreases the reliability of the parameter estimates.

Item focussed logistic trees (figure 7.8) find seven DIF items. In addition to the items detected by the other procedures as well, items 8 and 36 are classified as DIF items. For item 31, two splits are carried out according to the same variable, social status. This divides the observations into three groups. The item is most difficult for students with a very low social status, followed by students with a social status above -0.57. The middle group has the highest probability of solving the item correctly. Item 41 is split in the same way as split by *DIFtree* Rasch, but the interpretation implied by the resulting coefficients is slightly different. This shows again, that the parameter for students with a low social status and a migration background (-20.9 for IFLT vs. 6.48 for IFRT) is not very reliable. If one takes a closer look at the subgroup corresponding to the parameter estimate, almost none of the students was able to solve item 41 correctly in this subgroup. Thus, *DIFtree* detects a meaningful partition (students that are likely to fail at solving the item) even though parameters are difficult to estimate. The *DIFboost* procedure finds social status and language as DIF variables for item 41, but language and migration are correlated.

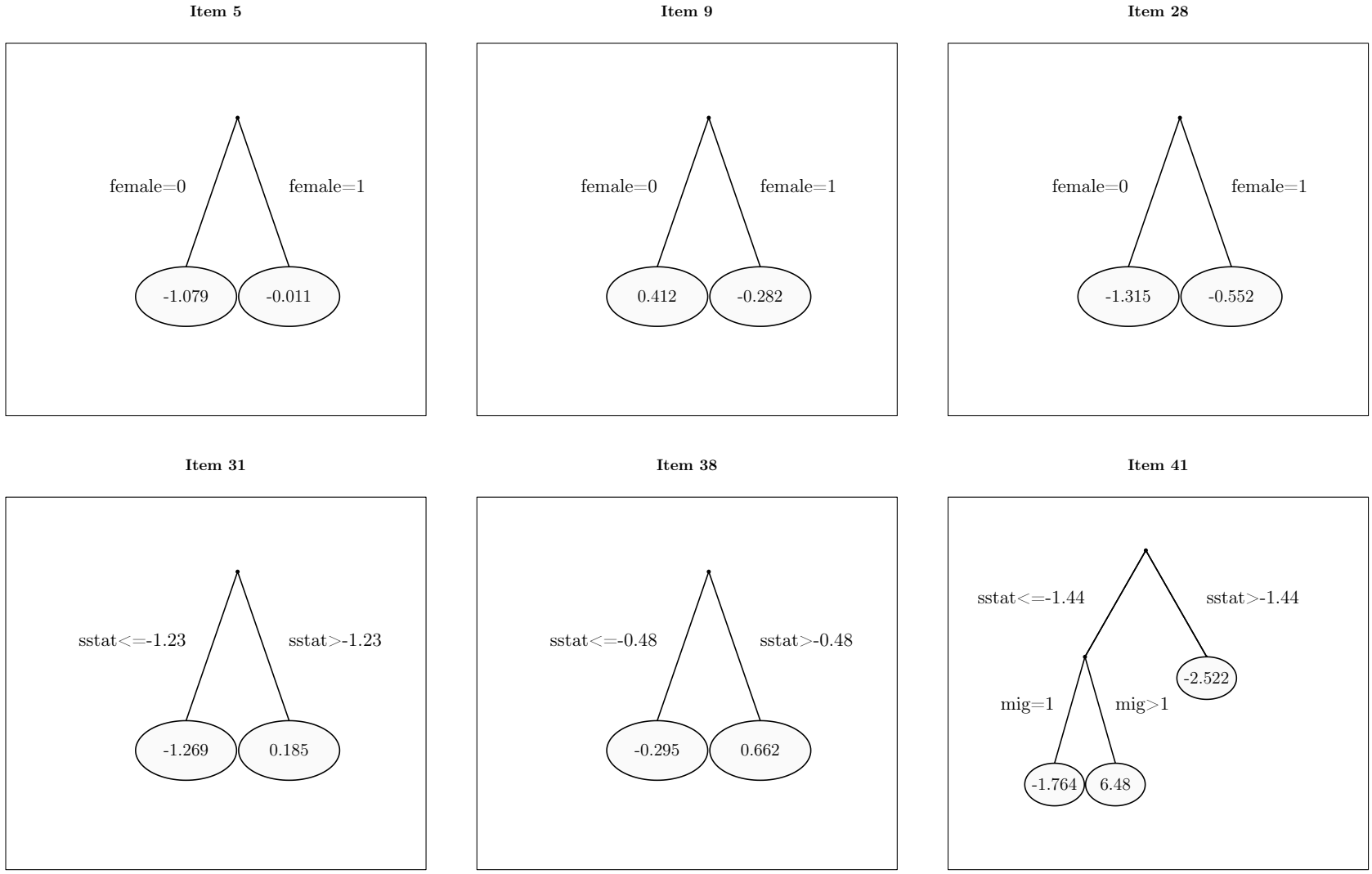

Figure 7.7.: Estimated item focussed Rasch trees

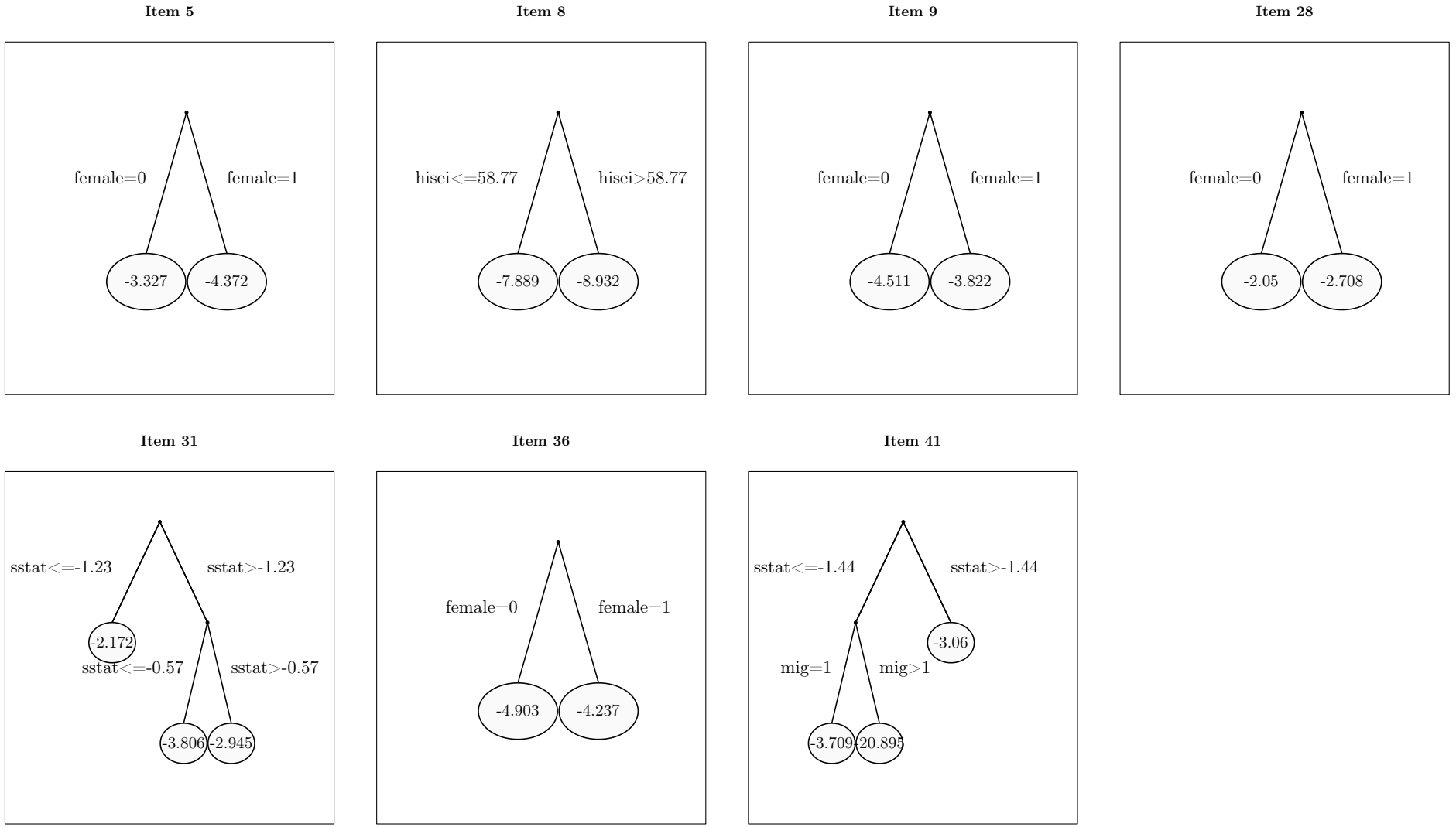

Figure 7.8.: Estimated item focussed logistic trees

7.4. Comparison of empirical results

Table 7.3 summarizes the findings of the empirical example. Two *DIFlasso* settings are not displayed, since the grouped/L2 *DIFlasso* finds no DIF items at all and the ungrouped/YL *DIFlasso* finds twenty DIF items, which does not seem plausible in comparison to the results of the other methods.

Item	DIFlasso		DIFboost	DIFtree		$\sum$ DIF
	ungr/YL	gr/L2		Rasch	logistic	classifications
3			x			1
5	x	x	x	x	x	**5**
8					x	1
9			x	x	x	3
12			x			1
28		x	x	x	x	**4**
31				x	x	2
36					x	1
38		x	x	x		**3**
41		x	x	x	x	**4**
# DIF items	1	4	7	6	7	

Table 7.3.: Overview of detected DIF items in the empirical example of the different methods

Overall, the results differ depending on which method was used for the detection of differential item functioning. Especially the sparseness in terms of how many items are classified as DIF items varies between the methods. The *DIFlasso* procedure leads to more sparse models than the other two procedures, except for the setting, where the regular lasso penalty is used together with the L2 type degrees of freedom. If the degrees of freedom for the BIC are calculated according to Yuan and Lin (2006), models are more restricted than using the L2 norm. *DIFboost* and logistic *DIFtree* both detect seven DIF items.
Items 5, 28, 38 and 41 are the items that are classified as DIF items by *DIFlasso*, *DIFboost* and *DIFtree*. Item 5 is the only item that is found by *DIFlasso* ungr/YL.
On the item level, the results are mostly in accordance with each other. DIF variables and direction of influence match, when the item is detected by more than one procedure. Group specific differences mostly correspond to the gender or the social status of the students. The biggest difference between *DIFtree* and the other methods then is the treatment of numerical covariates (with the latter finding binary splits for metric covariates in each step). During the simulation study, *DIFboost* and *DIFtree* gave better results than *DIFlasso*. This should be kept in mind for the interpretation of the empirical results as well.
As a conclusion, it seems advisable to apply more than one method to detect DIF and compare the results between the methods. If items are chosen as DIF items by more than one procedure, their classification as DIF items is more reliable than if an item is just chosen by one procedure as one out of many items.

8. Conclusion

This thesis presented and compared three methods developed for the detection of differential item functioning, including *DIFlasso*, an approach that estimates the DIF model using lasso penalization in order to determine DIF items. The second method, *DIFboost*, finds the model parameters related to DIF via boosting. *DIFtree* detects DIF via model-based recursive partitioning, where the DIF model or the logistic model can be chosen as underlying models. They have in common that they were developed to overcome the limitations of existing methods regarding the type and number of predictor variables that can be included. Also, they detect DIF not only on the global test level but on the item level, allowing a conclusion about which items exhibit group differences. The main difference between *DIFlasso/DIFboost* and *DIFtree* is the treatment of numerical variables. The first two mentioned methods assume a linear effect on the success probabilities here. *DIFtree* splits the predictor space into subregions, estimating one item difficulty in every subregion. Subgroups of the predictor variables do not have to be prespecified here (i.e. split points for metric variables are found by the procedure, starting with the most important variable at the root of the tree). Trees might also be more capable of representing interactions between the predictor variables.

Taking the aforementioned advantages and flexibility aspects into account, the application of the presented method might be useful for researchers and practitioners in addition to the well established methods introduced in chapter 2. The performance of the three methods in relation to these established methods was investigated in the originating papers Tutz and Schauberger (2015), Schauberger and Tutz (2016), Tutz and Berger (2016) and Berger and Tutz (2016). Therefore, the thesis was intended to provide additional information on the three methods, regarding their performance in relation to each other, both in simulations as well as using a practical data set assessing the mathematical abilities of 8th grade students in Austria.

The simulation study consisted of two different scenarios. In the first scenario, the data was generated according to the DIF model including five binary and metric covariates. In the second scenario, the data was again generated according to the DIF model but the three metric covariates were binarized corresponding to a fixed split point before success probabilities were calculated. From theoretical considerations, it was expected that in the first scenario, *DIFlasso* and *DIFboost* would give better results, whereas in the second scenario, the data structure should be better captured by *DIFtree*. In the strong and medium setting of scenario 1, all methods perform very well in terms of their detection rates and differences are small. Especially in the weak setting of scenario 1, *DIFboost* outperforms the other methods clearly, followed by *DIFtree*. The second scenario leads to lower true positive rates in general than the first scenario. In the strong DIF setting, *DIFtree* outperforms the other methods, as was expected. Surprisingly, in the other two settings of scenario 2, *DIFboost* leads to slightly higher true positive rates than *DIFtree*. In both scenarios, *DIFlasso* cannot compete with the other methods, especially when group differences are small. The *DIFtree* methodology with underlying logistic model performs slightly better than with underlying Rasch model and has the advantage that computation times are much lower. The differences between the methods regarding mean squared

errors of the person parameter and the group-specific item difficulty are small. The only exception is the mse of the group-specific item difficulty for item focussed Rasch trees, that has some outliers in a few of the simulation iterations. This increases the weaker the DIF effects are in scenario 2. Overall, *DIFboost* and *DIFtree* perform comparably well during the simulation study, whereas *DIFlasso* fails to detect DIF items, especially when the DIF effects are small.

For the practical example, data from the Austrian 8th grade assessment of education standards in mathematics was provided. It included information on the performance of 851 students on 48 test items as well as five socio-economic background variables. Same as during the simulation study, *DIFboost* and *DIFtree* give similar results, finding six to seven DIF items. Detected items and directions of influence (as can be read from the size and sign of the group parameters) mostly match. *DIFlasso* leads to more sparse model in terms of the number of detected DIF items. Here, results vary more over the different parameter settings of *DIFlasso*. *DIFlasso* with ungrouped lasso penalty and degrees of freedom according to Yuan and Lin (2006) finds the most DIF items (20), whereas *DIFlasso* using the group lasso penalty and degrees of freedom being the L2 norm of the group parameters finds no DIF items at all.

It should be noted that for categorical predictor variables, the results of *DIFlasso* depend on the chosen reference category. Choosing a different parameterization leads to different results. This is why, in opposite to metric or binary variables, categorical predictors should be included with care and keeping this effect in mind. Future research might be able to overcome this problem. In addition, for most of the parameters, the default values were handed over to the algorithms, meaning that there is also room for further explorations to see how choosing other values would influence the results. Even though *DIFboost* and *DIFtree* give similar results, it is advisable to apply different methods in order to find a set of stable DIF items.

List of Figures

List of Tables

Bibliography

Berger, M. and G. Tutz (2016). Detection of uniform and non-uniform differential item functioning by item focussed trees. *Journal of Educational and Behavioral Statistics 41*(6), 559–592.

BIFIE (2014). Daten zur Standardüberprüfung Mathematik, 8. Schulstufe, 2012. fdb-sp:bistM812I-item-16a.1. Salzburg:BIFIE.

Birnbaum, A. (1968). Some latent train models and their use in inferring an examinee's ability. In F. M. Lord and M. R. Novick (Eds.), *Statistical theories of mental test scores*, pp. 395–479. Addison-Wesley.

Breiman, L., J. Friedman, C. J. Stone, and R. A. Olshen (1984). *Classification and regression trees*. CRC press.

Friedman, J., T. Hastie, and R. Tibshirani (2000). Additive logistic regression: a statistical view of boosting. *The annals of statistics 28*(2), 337–407.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics 29*(5), 1189–1232.

Heugl, H., W. Peschek, M. Dangl, G. Jurkowitsch, M. Katzenberger, B. Kröpfl, F. Picher, E. Schneider, and R. Scheriau (2007). *Standards für die mathematischen Fähigkeiten österreichischer Schülerinnen und Schüler am Ende der 8. Schulstufe*. Institut für Didaktik der Mathematik. Alpen-Adria-Universität Klagenfurt.

Itzlinger-Bruneforth, U., T. Kuhn, and T. Kiefer (2016). Testkonstruktion. In C. Schreiner and S. Breit (Eds.), *Large-Scale Assessment mit R. Methodische Grundlagen der österreichischen Bildungsstandard-Überprüfung* . facultas.

Kiefer, T., T. Kuhn, and R. Fellinger (2016). Testdesign. In C. Schreiner and S. Breit (Eds.), *Large-Scale Assessment mit R. Methodische Grundlagen der österreichischen Bildungsstandard-Überprüfung* . facultas.

Kim, S.-H., A. S. Cohen, and T.-H. Park (1995). Detection of differential item functioning in multiple groups. *Journal of Educational Measurement 32*(3), 261–276.

Kuhn, J.-T. and T. Kiefer (2013). Optimal test assembly in practice. *Zeitschrift für Psychologie 221*(3), 190–200.

Lord, F. M. (1980). *Applications of item response theory to practical testing problems*. Routledge.

Magis, D., G. Raîche, S. Béland, and P. Gérard (2011). A generalized logistic regression procedure to detect differential item functioning among multiple groups. *International Journal of Testing 11*(4), 365–386.

Mantel, N. and W. Haenszel (1959). Statistical aspects of the analysis of data from retrospective studies. *Journal of the National Cancer Institute 22*(4), 719–748.

Meinshausen, N. and P. Bühlmann (2010). Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 72*(4), 417–473.

Millsap, R. E. and H. T. Everson (1993). Methodology review: Statistical approaches for assessing measurement bias. *Applied psychological measurement 17*(4), 297–334.

Penfield, R. D. (2001). Assessing differential item functioning among multiple groups: A comparison of three Mantel-Haenszel procedures. *Applied Measurement in Education 14*(3), 235–259.

Quinlan, J. R. (2014). *C4. 5: programs for machine learning*. Elsevier.

R Core Team (2015). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

Rasch, G. (1960). Probabilistic models for some intelligence and achievement tests. *Copenhagen: Danish Institute for Educational Research*.

Schauberger, G. and G. Tutz (2016). Detection of differential item functioning in Rasch models by boosting techniques. *British Journal of Mathematical and Statistical Psychology 69*(1), 80–103.

Schreiner, C. and S. Breit (Eds.) (2012). *Standardüberprüfung 2012 Mathematik, 8. Schulstufe - Bundesergebnisbericht. Salzburg.* Available at https://www.bifie.at/node/2489.

Strobl, C. (2012). *Das Rasch-Modell: Eine verständliche Einführung für Studium und Praxis*, Volume 2. Rainer Hampp Verlag.

Strobl, C., J. Kopf, and A. Zeileis (2015). Rasch trees: A new method for detecting differential item functioning in the Rasch model. *Psychometrika 80*(2), 289–316.

Swaminathan, H. and H. J. Rogers (1990). Detecting differential item functioning using logistic regression procedures. *Journal of Educational measurement 27*(4), 361–370.

Tutz, G. and M. Berger (2016). Item-focussed trees for the identification of items in differential item functioning. *Psychometrika 81*(3), 727—750.

Tutz, G. and G. Schauberger (2015). A penalty approach to differential item functioning in rasch models. *Psychometrika 80*(1), 21–43.

Yuan, M. and Y. Lin (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68*(1), 49–67.

A. Contents of enclosed CD

The enclosed CD includes a PDF-version of this thesis, the current gz archives of *DIFlasso*, *DIFboost* and *DIFtree*, that were used for the practical considerations, as well as five folders with the generated R-Code and graphics. The folders are named and structured as follows:

- **01_Intro_Plots**: contains the R-code for the generation of the ICC plots in chapter 2
- **02_SIM_1**: contains an R-script that generates the data of scenario 1 as well as an R-script where the different methods are applied under the different settings
- **03_SIM_2**: contains R-scripts for the data generation of scenario 2 as well as for the application of the different methods under the different settings
- **04_Nach_SIM** has two sub-folders:
  - R-Skripte: R-scripts for the calculation of error rates and mean squared errors, one for each scenario and method. Moreover, a script for the generation of the plots and the tables and a short script that extracts the relevant information from the large *DIFboost* simulation results for the further analysis, that needs to be run before the characteristic numbers can be calculated
  - TEX: contains the .tex-files, that will produce the graphics later in the latex file, using R-package *tikzDevice*
- **05_Bifie**
  - R-Skripte: contains R-scripts for the data preparation, the DIF analysis and the generation of the plots
  - PDF: contains PDF- (*raschtree* plot) and .tex-files of the graphics produced for the illustration of the empirical example

Statement

I hereby declare that this thesis is my own original work and that all sources have been acknowledged.

Munich, January 25, 2017

_____

Stephanie Hubert